

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

**Інститут прикладного системного аналізу
Кафедра математичних методів системного аналізу**

До захисту допущено:

В.о.завідувача кафедри

_____ Оксана ТИМОЩУК

«__» _____ 20__ р.

Дипломна робота

на здобуття ступеня бакалавра

за освітньо-професійною програмою «Системний аналіз і управління»

спеціальності 124 «Системний аналіз»

на тему: «Моделювання та прогнозування

фінансової стійкості компанії з використанням методів ІАД»

Виконала:

студентка IV курсу, групи КА-64

Черниш Злата Святославівна _____

Керівник: д.т.н., доцент кафедри ММСА,

Кузнєцова Н.В. _____

Консультант з економічного розділу:

к.е.н., доцент кафедри теоретичної та прикладної економіки _____

Шевчук О.А.

Консультант з нормоконтролю:

к.т.н., доцент кафедри ММСА

Коваленко А.Є. _____

Рецензент:

к.т.н., доцент кафедри СП

Безносик О.Ю. _____

Засвідчую, що у цій дипломній роботі немає
запозичень з праць інших авторів без
відповідних посилань.

Студентка _____

Київ – 2020 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Інститут прикладного системного аналізу
Кафедра математичних методів системного аналізу

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 124 «Системний аналіз»

Освітньо-професійна програма «Системний аналіз і управління»

ЗАТВЕРДЖУЮ

В.о.завідувача кафедри

_____ Оксана ТИМОЩУК

«__» _____ 20__ р.

ЗАВДАННЯ

на дипломну роботу студентці

Черниш Златі Святославівні

1. Тема роботи «Моделювання та прогнозування фінансової стійкості компанії з використанням методів ІАД», керівник роботи : Кузнєцова Наталія Володимирівна, д.т.н., доцент кафедри ММСА,

затверджені наказом по університету від «_25_» _травня_ 20_20_ р. №_1143-с_

2. Термін подання студентом роботи 12.06.2020 р.

3. Вихідні дані до роботи

4. Зміст роботи

5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо)

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

Економічний розділ	Шевчук О.А, к.е.н., доцент кафедри теоретичної та прикладної економіки	21.04.2020	25.05.2020
Нормконтроль	Коваленко А.Є., к.т.н., доцент кафедри ММСА	20.05.2020	11.06.2020

7. Дата видачі завдання 18.09.2019

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітка
1	Визначення із темою роботи	18.09.2019	Виконано
2	Написання теоретичної частини роботи. Збір та аналіз вхідних даних.	20.11.2019	Виконано
3	Ознайомлення з літературою. Визначення структури роботи на етапі планування.	25.02.2020	Виконано
4	Продовження опрацювання джерел. Написання теоретичної частини роботи. Експериментальне моделювання вхідних даних.	13.04.2020	Виконано
5	Продовження експериментального моделювання вхідних даних. Оформлення вступу, першого та другого розділів, списку використаних джерел. Розробка інтерфейсу користувача інформаційної системи.	18.05.2020	Виконано
6	Завершення роботи над економічним розділом роботи.	24.05.2020	Виконано
7	Підготовка роботи до передзахисту. Створення презентації.	29.05.2020	Виконано
8	Перевірка роботи на антиплагіат.	01.06.2020	Виконано
9	Перевірка роботи на нормконтроль.	11.06.2020	Виконано
10	Захист дипломної роботи.	19.06.2020	Виконано

Студент

Злата ЧЕРНИШ

Керівник

Наталія КУЗНЄЦОВА

РЕФЕРАТ

Дипломна робота: 112 с., 19 рис., 9 табл., 2 додатки, 23 джерела.

АР, АРКС, АРІКС, АРУГ, ЛІНІЙНА РЕГРЕСІЯ, МГУА, НЕЙРОННІ МЕРЕЖІ, ПРОГНОЗУВАННЯ, УАРУГ, ЧАСОВИЙ РЯД.

Об'єкт дослідження: вибірка даних, взятих із звітів про фінансові результати, а також фінансові коефіцієнти, розраховані на їх основі, що потребують подальшого аналізу із виявленням закономірностей та взаємозалежностей між елементами.

Предмет дослідження: методи та моделі прогнозування часових рядів: моделі авторегресії, авторегресії – ковзного середнього, авторегресії – інтегрованого ковзного середнього, авторегресії з умовною гетероскедастичністю, моделі лінійної регресії, моделі на основі нейронних мереж, моделі на основі методу групового урахування аргументів, а також згладжування та фільтрація часових рядів.

Мета дослідження: розробка та реалізація інформаційної системи для моделювання та прогнозування економічних та фінансових явищ та показників.

Результат дослідження: створення програмного продукту за допомогою мови програмування для статистичної обробки даних R у програмному середовищі R Studio. За допомогою розробленого програмного продукту побудовано та описано моделі для прогнозування дохідності компанії. Виконано огляд вітчизняної та зарубіжної літератури із сучасної економетрики, економічного та фінансового аналізу.

ABSTRACT

Batchelor's thesis: 112 p., 19 fig., 9 tab., 2 appendixes, 23 references.

AR, ARMA, ARIMA, ARCH, FORECASTING, GARCH, LINEAR REGRESSION, MGDH, NEURAL NETWORKS, TIME SERIES.

Object of the study: a sample of data taken from the income statements, alongside financial ratios calculated on the basis of their items, which require further analysis in order to identify patterns and correlation between the elements.

Subject of the study: time series forecasting methods and models such as autoregressive models, autoregressive - moving average models, autoregressive - integrated moving average models, autoregressive conditional heteroscedasticity models, linear regression models, models based on neural networks, models based on the method of group data handling, and time series smoothing and filtering.

The purpose of the study: an information system development and implementation for modeling and forecasting of economic and financial phenomena and indicators.

The result of the study: a software product developed in a programming language R for statistical data computing in the software environment R Studio. With the help of the developed software product models for the company's profitability forecasting are built and described. The review of Ukrainian and foreign literature on modern econometrics, economic and financial analysis was carried.

ЗМІСТ

ПЕРЕЛІК ПРИЙНЯТИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ	8
ВСТУП.....	10
РОЗДІЛ 1	11
ПРОГНОЗУВАННЯ ЯК МЕТОД ОЦІНКИ ФІНАНСОВОЇ ДІЯЛЬНОСТІ КОМПАНІЇ.....	11
1.1 Загальні поняття та завдання фінансового аналізу	11
1.2 Обґрунтування методів побудови фінансових прогнозів	13
1.3 Підготовка даних для подальшого аналізу та прогнозування.....	19
1.4. Висновки до розділу	22
РОЗДІЛ 2	23
ОГЛЯД ІСНУЮЧИХ ПІДХОДІВ І МОДЕЛЕЙ ДЛЯ ЗАДАЧІ ПРОГНОЗУВАННЯ	23
2.1 Моделі регресійного аналізу	23
2.2 Моделі авторегресії.....	26
2.2.1 Модель авторегресії	27
2.2.2 Модель ковзного середнього	28
2.2.3 Модель авторегресії ковзного середнього	29
2.2.4 Моделі авторегресії з умовною гетероскедастичністю.....	34
2.3 Моделі стохастичної волатильності.....	41
2.4 Моделі на основі нейронних мереж	42
2.5 Моделі на основі методу групового урахування аргументів	46
2.6 Критерії адекватності моделей.....	49
2.7 Критерії якості оцінок прогнозів.....	54
2.8 Висновки до розділу	55
РОЗДІЛ 3	57

ПОБУДОВА МОДЕЛЕЙ ДЛЯ ОЦІНЮВАННЯ ТА ПРОГНОЗУВАННЯ ДОХОДНОСТІ КОМПАНІЇ INTEL CORPORATION.....	57
3.1 Аналіз звіту про фінансові результати.....	57
3.2 Вибір програмного забезпечення для виконання обчислювальних експериментів.....	64
3.3 Побудова моделей та аналіз отриманих результатів.....	65
3.4 Висновки до розділу	75
РОЗДІЛ 4	77
ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ ПРОГРАМНОГО ПРОДУКТУ	77
4.1 Постановка завдання проектування	77
4.1.1 Обґрунтування функцій програмного продукту	77
4.1.2 Обґрунтування системи параметрів програмного продукту	80
4.1.3 Аналіз експертного оцінювання параметрів.....	82
4.2 Економічний аналіз варіантів розробки програмного продукту	85
4.3 Висновки до розділу	90
ВИСНОВКИ	92
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	93
ДОДАТОК А Лістинг програми.....	96
ДОДАТОК Б Ілюстративний матеріал для доповіді	102

ПЕРЕЛІК ПРИЙНЯТИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

AIC – Akaike information criterion

AR – autoregression

ARCH – autoregressive conditional heteroscedasticity

ARIMA – autoregressive - integrated moving average

BIC – Bayesian information criterion

GARCH – generalized autoregressive conditional heteroscedasticity

GMDH – group method of data handling

MAE – mean absolute error

MAPE – mean absolute percent error

MSE – mean squared error

NNAR – neural networks autoregression

SARIMA – seasonal autoregressive - integrated moving average

TheilU – Theil's U index of inequality

АКФ – автокореляційна функція

АР – авторегресія

АРІКС – авторегресія – інтегроване ковзне середнє

АРКС – авторегресія – ковзне середнє

АРУГ – авторегресійна умовна гетероскедастичність

ЕУАРУГ – експоненціальна узагальнена авторегресійна умовна
гетероскедастичність

ІС – інформаційна система

КС – ковзне середнє

МГУА – метод групового урахування аргументів

МНК – метод найменших квадратів

МСВ – модель стохастичної волатильності

НМ – нейронна мережа

ПП – програмний продукт

УАРУГ – узагальнена авторегресійна умовна гетероскедастичність

ФВА – функціонально-вартісний аналіз

ЧАКФ – часткова автокореляційна функція

ВСТУП

У зв'язку із постійними змінами у економічних системах постає необхідність у проведенні фінансового аналізу, що дозволяє керівництвам компаній та підприємств приймати ефективні та своєчасні управлінські рішення, діагностувати виникнення ознак банкрутства тощо. Одним із завдань в управлінні компанією є здійснення фінансового прогнозування, у результатах якого зацікавлені не лише власники та акціонери, а й інвестори, кредитори, клієнти і податкові органи.

У роботі описано та реалізовано основні етапи проведення моделювання та прогнозування фінансового стану компанії на основі моделей АРІКС/АРУГ, нейронних мереж та методів групового урахування аргументів.

У першому розділі наводиться огляд основних понять, завдання та основні методи та прийоми фінансового аналізу та прогнозування, розглядається процес підготовки даних для проведення подальшої роботи над ними, а також досліджується актуальність побудови прогнозів фінансових та економічних явищ і показників.

У другому розділі наводиться огляд моделей для задачі прогнозування, описуються критерії адекватності моделей та якості оцінок прогнозів.

Третій розділ присвячено аналізу основних показників, за допомогою яких можна оцінити фінансовий стан компанії, а також побудові та вибору оптимальних моделей, на основі яких здійснюється прогнозування. Також наводяться результати експериментів, пов'язаних із застосування методик та методів покращення якості моделей.

У четвертому розділі проведено функціонально-вартісний аналіз програмного продукту із метою оптимізації співвідношення його корисності та собівартості.

РОЗДІЛ 1

ПРОГНОЗУВАННЯ ЯК МЕТОД ОЦІНКИ ФІНАНСОВОЇ ДІЯЛЬНОСТІ КОМПАНІЇ

1.1 Загальні поняття та завдання фінансового аналізу

Невід’ємною частиною фінансової роботи компанії є фінансовий аналіз та оцінка її фінансового стану.

Фінансовий стан компанії – це сукупність показників, що характеризують його конкурентоспроможність (тобто платоспроможність або кредитоспроможність), використання капіталу та фінансових ресурсів, здатність виконувати зобов’язання перед державою або іншими підприємствами.

Можна виділити наступні завдання фінансового аналізу:

- визначення змін основних показників фінансового стану, як якісних, так і кількісних;
- визначення факторів, що впливають на фінансовий стан компанії;
- виявлення та ліквідація недоліків, що виникають внаслідок діяльності компанії;
- прогнозування положення компанії на ринку та зниження рівня невизначеностей, що виникають у результаті прийняття економічних рішень, направлених на майбутнє;
- виділення рекомендацій, направлених на покращення фінансових результатів компанії.

Одним із найважливіших інструментів фінансового аналізу є прогнозування фінансово-економічного стану компанії. Прогнозування – це основане на системі закономірностей та причинно-наслідкових зв’язків визначення стану та ймовірних шляхів розвитку об’єктів, процесів та явищ. Фінансове прогнозування – процес

розробки та реалізації прогнозів, метою яких є ефективна організація діяльності підприємства, передбачення наслідків прийняття певних рішень.

До основної мети фінансового прогнозування можна віднести оцінку очікуваного об'єму фінансових ресурсів, пошук джерел формування та напрямів ефективного використання фінансових ресурсів, формування рекомендацій щодо раціональної фінансової стратегії та тактики менеджменту компанії.

При прогнозуванні виділяють три види прогнозів:

- короткострокові - до 1 року – 1,5 роки, базується на якісній та кількісній оцінці змін попиту та пропозиції, об'ємів виробництва, конкурентоспроможності на ринку та індексів цін. При короткостроковому прогнозуванні враховують циклічні та нециклічні сезонні, часові, випадкові фактори;

- середньострокові – 4 – 6 років, характеризують планування вкладення капіталу та рух грошових коштів, виробниче планування, враховують циклічні та нециклічні фактори;

- довгострокові – більше 6 років, характеризують бізнес-планування номенклатури продукції, капіталовкладення, науково-дослідницьких робіт, місцезнаходження компанії.

Аналізу та прогнозування фінансових та економічних явищ та процесів присвячені роботи таких авторів, як Д. -Е. Бестенс, В. М. Ван ден Берга, К. Гренджера, Б. Джордана, Д. Дікі, К. Доугерті, Р. Інгла, Е. Сігела тощо. У наукових роботах цих авторів описані характеристики моделей та методів прогнозування фінансових показників та економічних явищ. Наприклад, Д.-Е. Бестенс досліджує застосування нейронних мереж для розв'язання задач аналізу кризових явищ, курсу акцій, явищ банкрутства банків та корпорацій. К. Греджер був одним із основоположників методів об'єднаних прогнозів для підвищення точності економічних прогнозів.

Проте більшість прикладів практичного застосування інструментарію прогнозування стосуються макроекономічних показників та явищ, світових фінансових та товарних ринків. Методи прогнозування фінансових показників компаній у своїй більшості базуються на елементарних моделях. Вплив зовнішнього середовища підприємств на його ключові фінансові показники наразі залишається недостатньо науково обґрунтованим.

1.2 Обґрунтування методів побудови фінансових прогнозів

Побудова фінансових прогнозів являє собою комплексний процес, у ході якого необхідно розв'язати широкий спектр соціально-економічних та науково технічних задач, для чого існує понад 150 різноманітних методів прогнозування.

За ступенем формалізації фінансові прогнози можна поділити на дві категорії: якісні та кількісні (рисунок 1.1).

Якісні методи прогнозування базуються на суб'єктивних оцінках та інтуїтивно-логічному мисленні. Їх використовують лише у тих випадках, коли на прогнозований процес впливає велика кількість факторів, усі з яких неможливо врахувати, або за наявності високого ступеня невизначеності та повної відсутності інформації щодо прогнозованого об'єкта.

Кількісні методи прогнозування ґрунтуються на математичних методах і підходах та залежно від факторів, що впливають на конкретні на фінансові показники, можуть поділятися на декілька підгруп.

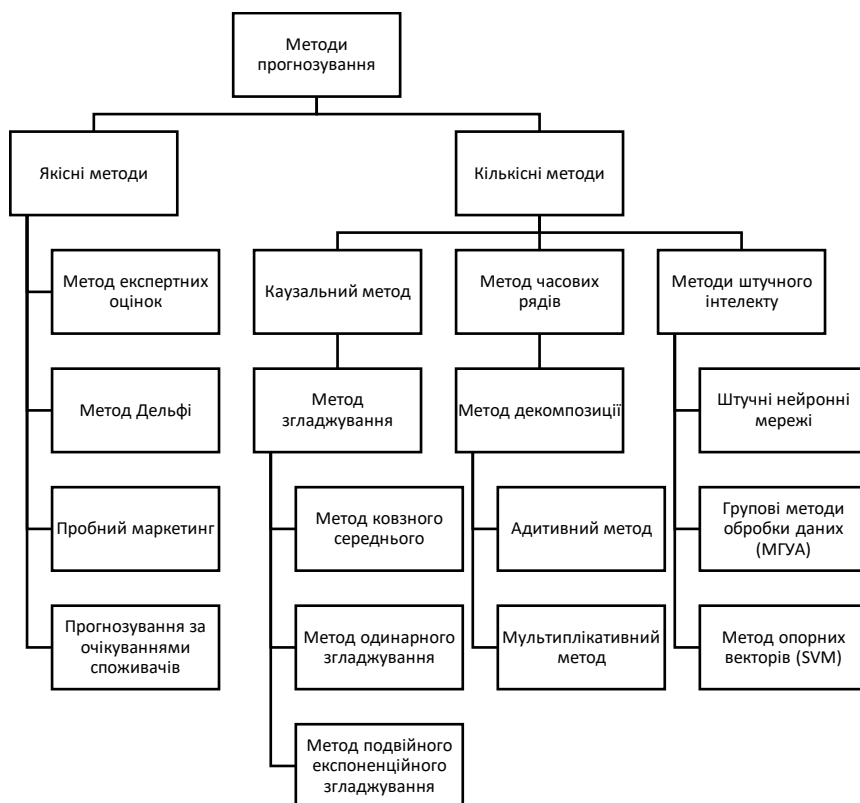


Рисунок 1.1 – Основні методи прогнозування

Методи аналізу часових рядів ґрунтуються на припущенні, що закон, за яким змінюються дані часового ряду, в минулому, є актуальним і в майбутньому. Часовим рядом називають масив даних, накопичених протягом певного періоду часу та розташованих у хронологічному порядку, для якого може бути характерна стійкі закономірності, що повторюються через окремі проміжки часу. Із метою виявлення таких закономірностей часовий ряд розкладають на окремі компоненти, серед яких можна виділити:

- трендова компонента – характеристика процесу зміни явища у висхідному або низхідному напрямі протягом певного проміжку часу;
- сезонна компонента – коливання навколо тренду, що повторюється через визначені проміжки часу, наприклад, тиждень, місяць, квартал тощо;
- періодична компонента - характеристика процесу зміни явища у висхідному або низхідному напрямі, що повторюється через проміжки часу, довжина яких

більша за один рік. У більшості випадків виникає через такі економічні явища, як інфляція або рецесія, тому частота появи закономірності даного типу не є стабільною;

– випадкова компонента – короткострокові, непостійні коливання, які не описуються закономірностями.

У процесі побудови прогнозу на основі моделей застосовують лише дві перші компоненти, що, за припущенням підходу з використанням часових рядів, є незалежними одна від одної.

Існує два типи моделей побудови прогнозу на основі часових рядів:

– мультиплікативна модель, елементами якої є компоненти часового ряду, виражені коефіцієнтами, добуток яких використовується для оцінки явища або показника;

– адитивна модель, за якої для оцінки явища розраховується сума усіх компонент.

У випадках, коли у формуванні прогнозованого явища не спостерігають стійких закономірностей, застосовують методи згладжування або фільтрації даних, що мають на меті заміну фактичних значень прогнозними, що мають менший рівень коливань, зашумленості та кількості зайвої інформації.

Одним із найпоширеніших фільтрів є фільтр Калмана, що застосовують для обчислення оптимальних оцінок залишків і побудови короткострокових прогнозів, заснованих на обраних моделях.

Алгоритм працює в два етапи.

Перший етап – екстраполяція. В основі роботи фільтра лежить модель простору станів, за якою будується прогноз поточного значення на основі попереднього:

$$x_k = F_k \widehat{x_{k-1}} + B_k u_k + w_k, \quad (1.1)$$

де x_k – прогност або оцінка поточного значення системи на k -ому кроці,

F_k – матриця залежності поточного стану системи від попереднього,

$\widehat{x_{k-1}}$ – значення системи на $k-1$ -ому кроці,

B_k – матриця керуючого впливу на систему,

u_k – вектор керування, або керуючого впливу,

w_k – доданий вектор гаусівського шуму.

Рівність (1.1) називається рівнянням переходу стану.

Коваріаційна матриця похибок оцінок параметрів $\widehat{x_k}$ у поточний момент часу P_k знаходиться із наступної рівності:

$$A_k = F_k \widehat{A_{k-1}} F_k^T + Q_k, \quad (1.2)$$

де $\widehat{A_{k-1}}$ – скорегована на попередньому кроці коваріаційна матриця помилок,

Q_k – коваріаційна матриця випадкового гаусівського шуму.

Другий етап – корегування значень системи, що складається із послідовності дій:

– визначення відхилення спостережуваного стану системи z_k від екстрапольованого значення x_k :

$$\overline{y_k} = z_k - H_k x_k, \quad (1.3)$$

де H_k – матриця, що показує залежність фактичного стану системи від спрогнозованих даних,

- ключовий етап алгоритму: визначення оптимальної матриці коефіцієнтів посилення Калмана K_k :

$$K_k = A_k H_k^T (H_k A_k H_k^T + R_k)^{-1}, \quad (1.4)$$

де R_k – коваріаційна матриця шуму системи;

- корегування оцінки стану системи:

$$\widehat{x}_k = \widehat{x}_{k-1} + K_k \overline{y}_k, \quad (1.5)$$

де \widehat{x}_k та \widehat{x}_{k-1} – скореговані значення на k -ому та $k-1$ -ому к роках. Тоді корегування дисперсія стану системи виглядає наступним чином:

$$\widehat{A}_k = (1 - K_k H_k) A_k, \quad (1.6)$$

Фільтр Калмана досить простий в реалізації, він автоматично підбирає посилення залежно від помилки прогнозу, причому це добре працює навіть в умовах шуму. Завдяки покроковій природі алгоритму, він може в реальному часі відстежувати стан спостережуваного процесу. Фільтр застосовують і в постобробці даних, коли є можливість враховувати всі результати вимірювань, змінювати початкові умови, тим самим підбираючи оптимальні коефіцієнти.

На практиці також досить поширеним є застосування цифрових фільтрів. Цифрова фільтрація – це система, що здійснює перетворення дискретного набору

вхідних даних x_t у дискретний набір вихідних даних y_t за допомогою лінійного співвідношення вигляду $y_t = \sum_{i=-\infty}^{+\infty} \alpha_i x_{t-i}$.

Цифрові фільтри використовують для розв'язання задач стиснення та згладжування даних, усунення трендів, поділ часових рядів на компоненти, пробне оцінювання спектрів.

Виділяють наступні види цифрових фільтрів:

1. Фільтр ковзного середнього – фільтр із скінченною кількісною характеристикою (на відфільтроване значення впливають лише N останніх значень), що оснований на розрахунку середнього на певному ковзному інтервалі:

$$y_t = \frac{1}{N} \sum_{i=0}^{N-1} x_{t+i}, \quad (1.7)$$

де x_t – вхідний сигнал,

y_t – вихідний сигнал,

N – кількість точок, що використовується для ковзного середнього.

2. Експоненціальне згладжування – фільтр, для якого на відфільтроване значення впливають усі попередні значення (з нескінченною імпульсною характеристикою). Перевагами фільтру є простота та здатність усувати великі скачки, а також відсутність необхідності зберігати у пам'яті декілька попередніх значень.

Записують експоненціальний фільтр за допомогою рівняння:

$$y_t = \theta x_t + (1 - \theta)y_{t-1}, \quad (1.8)$$

де x_t – обчислене значення у момент t ,

y_t – відфільтроване значення у момент t ,

θ – коефіцієнт фільтрації, що приймає значення від 0 до 1.

Серед експоненціальних фільтрів виділяють подвійний експоненціальний фільтр, що застосування якого еквівалентне двом фільтрам першого порядку та нелінійний фільтр, для якого $\theta = \min[1, \frac{x_t - y_{t-1}}{R}]$, R – коефіцієнт налаштування фільтра.

3. Поліноміальний фільтр.

4. Медіанний фільтр, що використовує масив N останніх значень, відсортованих за зростанням. Фільтрованим є значення, розташоване посередині відсортованого списку. На початку фільтрації кількість значень масиву менша за N , а потім збільшується із часом.

1.3 Підготовка даних для подальшого аналізу та прогнозування

Процедура обробки даних є важливим завданням, розв'язання якого необхідне для їх використання при побудові якісних моделей. Необроблені дані часто бувають спотвореними, зашумленими та неузгодженими, містять пропущені значення.

Підготовка даних включає наступні етапи:

1. Нормалізація даних та їх корегування дозволяють виявити та зменшити розміри шумів та викидів.
2. Дискретизація даних – перетворення неперервних характеристик у категоріальні для можливості застосування деяких методів машинного навчання.
3. Заповнення пропусків

На практиці дані часто бувають пошкодженими та містять ряд похибок, що негативно впливає на якість побудованої моделі. Найбільш поширеними проблемами, пов'язаними з якістю вхідних даних, є:

- пропущені значення;
- шуми та викиди;
- аномальні значення;
- дублікати даних.

Дані низької якості можуть бути спричинені рядом факторів, наприклад, неправильністю введення, неотримання даних та їх недостатньою обробкою, спотворенням даних тощо.

Існують різні підходи до розв'язання задачі заповнення пропусків даних.

Ігнорування пропущених значень.

Цей метод застосовують у випадках, коли у вибірці наявна незначна кількість об'єктів з пропущеними даними. Результатом методу є зменшення обсягу вибірки і запобігання спотворення даних шляхом заміни пропусків. Альтернативним методом є видалення пропущених значень, що також достатньо дієвий при роботі з вибірками великих обсягів.

Заповнення середнім або найбільш частим значенням.

Суть методу полягає в тому, що у випадку необхідності заповнення кількісних характеристик застосовують середнє значення ознаки, у випадку якісних – моду. Метод враховує наявні дані та усереднює їх, проте заповненням середнім значення послаблює міри асоціації, такі як коефіцієнти коваріації та кореляції, оскільки заповнені дані не корелюють із вихідними змінними. Тому цей метод є одним з найменш ефективних і на практиці його часто намагаються уникати.

Заміна за допомогою методу найближчих сусідів.

Метод ґрунтується на гіпотезі про те, що об'єкт належить до того класу, до якого належить більшість його «сусідів», під якими розуміють об'єкти, близькі до даного за певними ознаками. Проте для його реалізації необхідно визначити відстань між об'єктами, тобто наскільки вони є близькими один до одного, а отже, у просторі ознак об'єктів необхідно ввести певну метрику.

Заміна за допомогою методу випадкового лісу.

Випадковий ліс – метод машинного навчання, що базується на множині дерев прийняття рішень – ансамблі. Метод є універсальним і працює як з неперервними, так і категоріальними ознаками.

Оскільки серед ознак, за якими проводиться навчання, є пропущені значення, необхідно їх спочатку замінити за допомогою одного з найпростіших методів заповнення пропусків.

Наприклад, можна здійснити заповнення за допомогою заміни середнім значенням, а в якості алгоритма для уточнення пропусків використати випадковий ліс. Для кожної ознаки, що має пропущені значення, проводиться навчання вибірці об'єктів, що не мають пропусків за даною ознакою, для решти об'єктів на значення даної ознаки замінюють за допомогою навчально алгоритму. Ця процедура повторюється ітеративно до тих пір, поки не буде досягнуто збіжності або максимального числа ітерацій.

Заміна за допомогою бустингу.

Бустинг - це техніка побудови ансамблів, в якій класифікатори будують не незалежно, а послідовно.

В основі методи лежить ідея про те, що наступна модель буде навчатися на помилках попередньої. Вони із неоднаковою ймовірністю з'являються в наступних моделях, і частіше з'являються ті, що дають найбільшу помилку. Класифікатори можуть бути обрані з широкого асортименту моделей, наприклад, дерева рішень, регресія, класифікатори тощо. Оскільки їх навчання проводиться на помилках,

скоєних попередніми, потрібно менше часу для того, щоб отримати реальну відповідь. Але необхідно обережно підходити до вибору критерію зупинки, інакше це може привести до перенавчання. Прикладом бустингу є градієнтний бустинг.

1.4. Висновки до розділу

Прогнозування є одним із найбільш складних інструментів фінансового аналізу.

Найпростішим видом прогнозу є розрахунок очікуваного прибутку за різної зміни факторів, що на нього впливають, наприклад, операційних витрат та доходів, суми сплачених податків, амортизації тощо. Складнішим є комплексний прогноз фінансового стану компанії, для якого необхідно скласти прогнозні фінансові документи – прогнози доходів та витрат, прогнозний баланс.

Існує широке коло методичних підходів до прогнозування. Вибір прогнозуючого апарату і вміле його використання з метою прогнозування є досить комплексною проблемою, про що свідчить відсутність єдиного загальновизнаного набору конкретних методик і процедур прогнозування, адже у кожного методу є ряд своїх переваг та недоліків. Так, для якісних методів прогнозування характерна низька узгодженість точок зору експертів, виникнення суперечностей та пресингу усередині групи, і як наслідок, їх низька надійність. Тому для розв'язання практичних задач частіше застосовують більш формалізовані підходи, і орієнтуються на кількісні методи в управлінні фінансовими ресурсами або комбінацію якісних та кількісних методів прогнозування.

РОЗДІЛ 2

ОГЛЯД ІСНУЮЧИХ ПІДХОДІВ І МОДЕЛЕЙ ДЛЯ ЗАДАЧІ ПРОГНОЗУВАННЯ

2.1 Моделі регресійного аналізу

Будь-який економічний процес часто можна представити за допомогою певних кількісних характеристик, між якими необхідно встановити залежність та представити її в аналітичній формі, тобто через рівняння регресії. У такому випадку зв'язок між певними факторами (наприклад, ресурсів, що вкладені у дослідницьку діяльність підприємства, ціни на акції) та параметром, що досліджується (прибуток компанії), може бути показаний не лише у вигляді графіка, а й записаний з використанням емпіричної формули та побудований прогноз цільової змінної на основі набору вхідних ознак. Дослідження такої ситуації є задачею регресійного аналізу.

Регресія – функціональна залежність математичного сподівання залежної змінної від однієї або декількох інших пояснювальних змінних.

Серед регресійних моделей можна виділити:

- однопараметричні моделі $y = f(x)$ (що залежать від однієї змінної):
 - лінійні виду $y = \beta_0 + \beta_1 x$;
 - нелінійні (експоненційна, степенева, квадратична, логістична крива тощо);
- багатопараметричні моделі $y = f(x_1, x_2, \dots, x_n) + \varepsilon$ (залежність від декількох змінних):
 - лінійні виду $y = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n$, де x_1, x_2, \dots, x_n – незалежні;
 - моделі нелінійні за змінними та параметрами.

Лінійна множинна регресія.

Лінійна множинна регресія представлена наступним чином:

$$y = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n + \varepsilon, \quad (2.1)$$

де y — залежна змінна,

x_1, \dots, x_n — незалежні змінні,

ε — випадкова похибка.

Для кожного i -го спостереження залежність між змінними можна визначити за формулою:

$$y_i = \beta_0 + \beta_1 x_{i1} + \dots + \beta_n x_{in} + \varepsilon_i, \quad (2.2)$$

або у матричному вигляді

$$y = X\beta + \varepsilon, \quad (2.3)$$

де $y = [y_1, \dots, y_n]^T$ — вектор значень незалежної змінної,

$$X = \begin{bmatrix} 1 & x_{11} & x_{12} & \dots & x_{1n} \\ 1 & x_{21} & x_{22} & \dots & x_{2n} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & x_{p1} & x_{p2} & \dots & x_{pn} \end{bmatrix} \text{ — матриця незалежних змінних,}$$

$\beta = [\beta_1, \dots, \beta_n]^T$ — вектор коефіцієнтів моделі регресії,

$\varepsilon = [\varepsilon_1, \dots, \varepsilon_n]^T$ — вектор випадкових похибок.

Розглянемо умови та припущення, що лежать в основі регресійного аналізу.

1. Випадкова похибка ε_i не повинна мати систематичного зміщення. Ця умова виконується автоматично, якщо у рівняння регресії включити константну величину.

Залежні змінні x_1, \dots, x_n є не випадковими величинами, у протилежному випадку оцінки коефіцієнтів регресії є зміщеними або неконзистентними. Порушення цих умов може бути пов'язаним із використанням лагових змінних – пояснювальних змінних, взятих у регресійній моделі із запізненням у часі.

2. Математичне сподівання похибки в кожному спостереженні $E(\varepsilon_i) = 0$.

Дисперсія похибки $D(\varepsilon_i) = \sigma^2$ в кожному випадку є константою, тобто виконується умова гомоскедастичності – незалежність дисперсії випадкової похибки від номера спостереження.

Залежність випадкової дисперсії від номера спостереження називається гетероскедастичністю (“неоднакове розсіювання”), тобто коли $D(\varepsilon_i) = \sigma_i^2$ (рисунок 2.1).

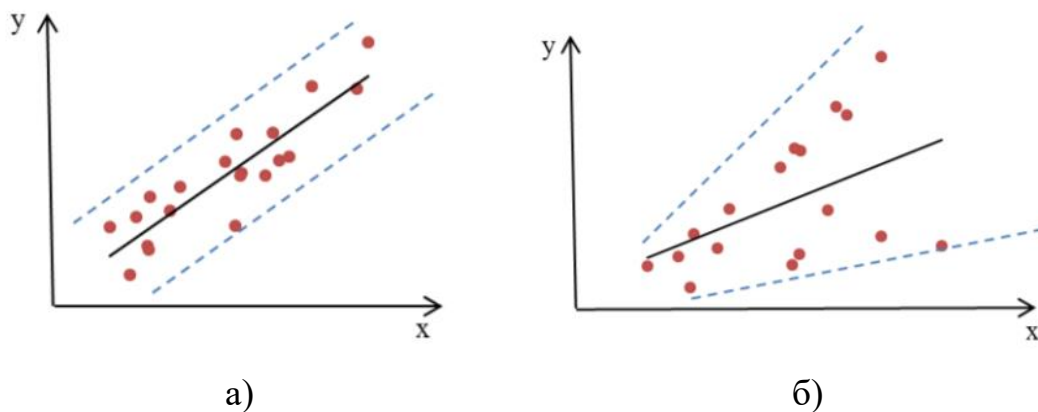


Рисунок 2.1 - Діаграма розсіювання для випадку гомоскедастичності (а) та гетероскедастичності (б)

3. Випадкові похибки ε_i та ε_j мають бути некорельованими (статистично незалежними) між собою: $E(\varepsilon_i, \varepsilon_j) = 0 \forall i \neq j$. У протилежному випадку коефіцієнти лінійної регресії, отримані за допомогою методу найменших квадратів, хоча і незміщеними, але неефективними, спостерігається явище автокореляції.

4. Випадкова похибка ε_i є нормально розподіленою величиною: $\varepsilon_i \sim N(0, \sigma^2)$.

Теорема Гаусова-Маркова.

Якщо виконуються припущення 1-4 регресійного аналізу, оцінки β_0 та β_1 , одержані за допомогою МНК є:

- незміщеними: $E(\beta_0) = \beta_0, E(\beta_1) = \beta_1$;
- конзистентними: $\lim_{n \rightarrow \infty} D(\beta_0) = \lim_{n \rightarrow \infty} D(\beta_1) = 0$;
- ефективними:

5. Ранг матриці значень залежних змінних $\text{rang } X = n + 1 < p$, тобто матриця є невиродженою.

2.2 Моделі авторегресії

Моделі авторегресії широко застосовуються для опису стаціонарних процесів в економічних дослідженнях.

Побудова авторегресійних моделей в економіці заснована на такому важливому властивості рядів економічних явищ і процесів, як взаємозалежність рівнів одного і того ж ряду один від одного. Умова нормальності розподілу ряду при побудові їх економетричних моделей не є обов'язковою.

Застосування авторегресійних моделей широко поширене для опису стаціонарних випадкових процесів. Визначною характеристикою стаціонарних часових рядів є те, що їх розвиток відбувається в незмінних стабільних умовах без вираженої тенденції, тому ймовірнісні властивості рядів залишаються незмінними від часу. Функції розподілу стаціонарних динамічних рядів також зберігають сталість при зсуві часу.

2.2.1 Модель авторегресії

У моделі авторегресії поточне значення процесу $y = y_t, t \geq 1$ можна представити у вигляді лінійної комбінації скінченного числа попередніх значень процесу та білого шуму ε_t , тобто:

$$y_t = \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \dots + \alpha_p y_{t-p} + \varepsilon_t, \quad (2.4)$$

де ε_t не корельовані із лагами y_t .

Така модель називається авторегресією порядку p (або $AR(p)$).

Введемо лаговий оператор L , що визначається наступним чином:

$$\alpha(L) = 1 - \alpha_1 L - \alpha_2 L^2 - \dots - \alpha_p L^p. \quad (2.5)$$

Тоді модель авторегресії можна записати:

$$\varepsilon_t = y_t(1 - \alpha_1 L - \alpha_2 L^2 - \dots - \alpha_p L^p) = \alpha(L)y_t. \quad (2.6)$$

Нехай маємо авторегресію першого порядку $AR(1)$:

$$y_t = \alpha_1 y_{t-1} + \varepsilon_t, \quad (2.7)$$

де ε_t —білий шум, некорельований із y_t , тобто $cov(\varepsilon_t, y_{t-1}) = 0, \alpha_1 = \alpha$.

Тоді записавши дисперсію обох частин, отримаємо $\sigma_y^2 = \alpha\sigma_y^2 + \sigma_\varepsilon^2$, тоді умовою стаціонарного процесу буде $|\alpha| < 1$.

Для авторегресійного процесу другого порядку AR(2) $y_t = \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \varepsilon_t$ умовою стаціонарності є $\alpha_1 \pm \alpha_2 < 1, -1 < \alpha_2 < 1$.

2.2.2 Модель ковзного середнього

Іншою поширеною моделлю є модель ковзного середнього (КС або moving average(MA)), що описує лінійну залежність процесу y_t від q його попередніх значень ε :

$$y_t = \varepsilon_t - \beta_1 \varepsilon_{t-1} - \dots - \beta_q \varepsilon_{t-q}. \quad (2.8)$$

Позначивши оператор ковзного середнього через $\beta(L)$, запишемо модель так:

$$y_t = \beta(L)\varepsilon_t, \quad (2.9)$$

де $\beta(L) = 1 - \beta_1 L - \beta_2 L^2 - \dots - \beta_q L^q$.

Легко побачити, що процес КС(q) є стаціонарним без накладання обмежень на параметри β .

Математичне сподівання процесу $E(y_t) = 0$, а дисперсія $\sigma_0^2 = (1 + \beta_1^2 + \dots + \beta_q^2)\sigma_\varepsilon^2 = C\sigma_\varepsilon^2$. Інші моменти другого порядку також не залежать від часу.

2.2.3 Модель авторегресії ковзного середнього

На практиці іноді буває доцільне введення у модель як елементів авторегресії, так і складову ковзного середнього для того, щоб оцінити характеристики часового ряду, використовуючи меншу кількість параметрів. Такий процес отримав назву змішаним процесом авторегресії – ковзного середнього і позначається АРКС(p,q) або ARMA(p,q):

$$y_t = \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \dots + \alpha_p y_{t-p} + \varepsilon_t - \beta_1 \varepsilon_{t-1} - \dots - \beta_q \varepsilon_{t-q}, \quad (2.10)$$

$$y_t(1 - \alpha_1 L - \alpha_2 L^2 - \dots - \alpha_p L^p) = \varepsilon_t(1 - \beta_1 L - \beta_2 L^2 - \dots - \beta_q L^q), \quad (2.11)$$

$$\beta(L)\varepsilon_t = \alpha(L)y_t, \quad (2.12)$$

де $\alpha(L)$ – оператор авторегресії,

$\beta(L)$ – оператор ковзного середнього.

У результаті досліджень Бокс і Дженкінс виділили клас нестационарних рядів, які за допомогою процедури взяття послідовних різниць можна привести до виду АРКС. Якщо після взяття d послідовних різниць ряд зводиться до стаціонарного вигляду, для побудови прогнозу застосовують комбіновану авторегресійну модель із проінтегрованим ковзним середнім – АРІКС(p,d,q) або ARIMA(p,d,q).

Методологія підбору моделі АРІКС для конкретного ряду спостережень складається із чотирьох етапів:

- вибір моделі, що найбільш відповідає реальному процесу;
- оцінювання моделі – використання регресійних методів для отримання оцінених параметрів, включених у модель;

- тестування моделі – перевірка адекватності із використанням тестів на нормальний розподіл (тест Жака- Бера), автокореляцію залишків (тест Дарбіна-Уотсона), на якість специфікації моделі (F-тест);
- використання моделі для побудови прогнозу.

Оцінювання параметрів моделі.

Щоб оцінити невідомі параметри моделі АРКС(p,q), застосовують метод максимальної правдоподібності або звичайний або нелінійний метод найменших квадратів.

За допомогою МНК параметри моделі розраховуються таким чином, щоб сума квадратів залишків була мінімальною.

Розглянемо модель авторегресії АР(p): $y_t = \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \dots + \alpha_p y_{t-p} + \varepsilon_t$, де $E(\varepsilon_t, y_{t-i}) = 0, i = 1, \dots, p$.

Тобто залишки та факторні змінні одночасно некорельовані, застосований до моделі МНК забезпечує конзистентні оцінки. А отже, оцінювання авторегресійної моделі є подібним до оцінювання лінійної регресії із лаговою змінною.

Розглянемо модель КС(1): $y_t = \varepsilon_t + \beta_1 \varepsilon_{t-1}$, де ε_{t-1} не є спостережуваним, тому метод регресії не може бути застосований. Запишемо ε_{t-1} як функцію від спостережуваних y_t :

$$\varepsilon_{t-1} = \sum_{j=0}^{\infty} (-\beta)^j y_{t-j-1}. \quad (2.13)$$

Тоді використовуючи МНК, маємо:

$$S(\beta) = \sum_{t=2}^T (y_t - \beta \sum_{j=0}^{\infty} (-\beta)^j y_{t-j-1})^2 \quad (2.14)$$

$$\hat{S}(\beta) = \sum_{t=2}^T (y_t - \beta \sum_{j=0}^{t-2} (-\beta)^j y_{t-j-1})^2 \quad (2.15)$$

Якщо $T \rightarrow \infty$, $S(\beta) \approx \hat{S}(\beta)$, мінімізація наближеної суми квадратів $S(\beta)$ дає конзистентну і асимптотично нормальну оцінку $\hat{\beta}$.

Прогнозування за допомогою АРІКС.

Для моделі АРІКС значення незалежної змінної прогнозується для деякого моменту $t + 1$ (у майбутньому), при цьому значення лагів критеріальної змінної можна вважати випадковими або фіксованими. Прогнозним значенням можна вважати функцію залежності від інформаційної множини, що містить лагові значення та попередні значення процесу y_t . Вибір цієї функції базується на мінімізації умовного математичного сподівання квадрату похибки прогнозу.

Нехай маємо модель КС(q): $y_t = \varepsilon_t + \beta_0 + \beta_1 \varepsilon_{t-1} + \dots + \beta_q \varepsilon_{t-q}$, для якої припускаємо, що коефіцієнти є визначеними, а також існують значення y_t , $t \in [1, T]$.

Прогнозом для моменту $T + 1$ вважається наступна рівність:

$$\begin{aligned} \widehat{y_{T+1}} &= E[y_{T+1} | y_1, \dots, y_T] = \\ &= E[\varepsilon_{T+1} + \beta_0 + \beta_1 \varepsilon_T + \dots + \beta_q \varepsilon_{T-q+1} | y_1, \dots, y_T] = \\ &= \beta_0 + \beta_1 E[\varepsilon_T | y_1, \dots, y_T] + \dots + \beta_q E[\varepsilon_{T-q+1} | y_1, \dots, y_T]. \end{aligned} \quad (2.16)$$

Можна побачити, що $E[\varepsilon_T | y_1, \dots, y_T] = y_T - \widehat{y_T}$, і так далі, що є різницею між реальним і прогнозованим значеннями, тоді

$$\widehat{y_{T+1}} = \beta_0 + \beta_1(y_T - \widehat{y_T}) + \\ + \beta_2(y_{T-1} - \widehat{y_{T-1}}) \dots + \beta_q(y_{T-q+1} - \widehat{y_{T-q+1}}). \quad (2.17)$$

При $h \geq q + 1$ прогноз для моменту h дорівнює $\widehat{y_{T+h}} = \beta_0$.

Дисперсія похибки $y_{T+1} - \widehat{y_{T+1}}$ на один крок уперед виглядає наступним чином:

$$\text{var}[y_{T+1} - \widehat{y_{T+1}} | y_1, \dots, y_T] = \\ E[(\varepsilon_{T+1} + \beta_0 + \beta_1\varepsilon_T + \dots + \beta_q\varepsilon_{T-q+1} - \beta_0 - \beta_1\varepsilon_T - \dots - \beta_q\varepsilon_{T-q+1}) | y_1, \dots, y_T]^2 = \\ = \sigma_\varepsilon^2. \quad (2.18)$$

Дисперсія похибки прогнозу на два кроки записується так:

$$\text{var}[y_{T+2} - \widehat{y_{T+2}} | y_1, \dots, y_T] = (1 + \beta_1^2)\sigma_\varepsilon^2, \quad (2.19)$$

на h кроків:

$$\text{var}[y_{T+h} - \widehat{y_{T+h}} | y_1, \dots, y_T] = (1 + \beta_1^2 + \dots + \beta_h^2)\sigma_\varepsilon^2, \quad (2.20)$$

при $h < q$.

При $h \geq q$ дисперсія похибки прогнозу рівна дисперсії процесу y_t .

Розглянемо модель стаціонарного процесу $AP(p)$: $y_t = \alpha_0 + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \dots + \alpha_p y_{t-p} + \varepsilon_t$.

Для одержання прогнозу на один крок уперед запишемо:

$$\begin{aligned}\widehat{y_{T+1}} &= E[y_{T+1}|y_1, \dots, y_T] = \\ &= E[\alpha_0 + \alpha_1 y_T + \alpha_2 y_{T-1} + \dots + \alpha_p y_{T-p+1} + \varepsilon_T | y_1, \dots, y_T] = \alpha_0 + \alpha_1 y_T + \alpha_2 y_{T-1} + \\ &\quad + \dots + \alpha_p y_{T-p+1}.\end{aligned}\tag{2.21}$$

Для другого кроку:

$$\begin{aligned}E[y_{T+2}|y_1, \dots, y_T] &= \\ &= E[\alpha_0 + \alpha_1 y_{T+1} + \alpha_2 y_T + \dots + \alpha_p y_{T-p+2} + \varepsilon_T | y_1, \dots, y_T].\end{aligned}\tag{2.22}$$

Математичне сподівання білого шуму ε є нульовим, а $E[y_T | y_1, \dots, y_T] = y_T$, $E[y_{T-1} | y_1, \dots, y_T] = y_{T-1}$ і т.д.

Для АР(p) маємо:

$$\widehat{y_{T+1}} = \alpha_0 + \alpha_1 y_T + \alpha_2 y_{T-1} + \dots + \alpha_p y_{T-p+1}.\tag{2.23}$$

А $y_{T+1} = \alpha_0 + \alpha_1 y_T + \alpha_2 y_{T-1} + \dots + \alpha_p y_{T-p+1} + \varepsilon_{T+1}$, і через неможливість прогнозувати ε_t , дисперсія похибки прогнозу, аналогічно дисперсії ковзного середнього рівна σ_ε^2 .

Прогнозування на два кроки уперед:

$$\widehat{y_{T+2}} = \alpha_0 + \alpha_1 y_{T+1} + \alpha_2 y_T + \dots + \alpha_p y_{T-p+2} =$$

$$= \alpha_1(\alpha_0 + \alpha_1 y_T + \alpha_2 y_{T-1} + \dots + \alpha_p y_{T-p+1} + \varepsilon_{T+1}) + \alpha_2 y_T + \dots + \alpha_p y_{T-p+2}, \quad (2.24)$$

$$y_{T+2} = \alpha_0 + \alpha_1 y_{T+1} + \alpha_2 y_T + \dots + \alpha_p y_{T-p+2} + \varepsilon_{T+2} = \alpha_0 + \alpha_1 * \\ * (\alpha_0 + \alpha_1 y_T + \dots + \alpha_p y_{T-p+1} + \varepsilon_{T+1}) + \alpha_2 y_T + \dots + \alpha_p y_{T-p+1} + \\ + \varepsilon_{T+2}. \quad (2.25)$$

Дисперсія похибки прогнозу на 2 кроки аналогічно рівна $(1 + \beta_1^2)\sigma_\varepsilon^2$.

Отже, для прогнозу на h кроків отримаємо:

$$\widehat{y_{T+h}} = \alpha_0 + \sum_{i=0}^p \gamma_i \widehat{y_{T+h-i}}. \quad (2.26)$$

Для моделі АРКС(p,q) необхідно додати отримані результати.

2.2.4 Моделі авторегресії з умовною гетероскедастичністю

Для моделювання волатильності доходності фінансових активів на практиці часто використовують моделі умовної гетероскедастичності, її модифікації та узагальнення.

Авторегресійна умовно гетероскедастична модель (АРУГ) або AutoRegressive Conditional Heteroscedasticity (ARCH) уперше згадується у статті Роберта Енгла та

полягає в тому, що дисперсія вільного члену ε_t у деякий момент часу t залежить від квадратів вільних членів попередніх моментів.

Позначимо за $y = y_t, t \geq 1$ випадковий процес, що є гаусівською моделлю, де $y_t = \sigma_t \varepsilon_t$, $\varepsilon = \varepsilon_t, t \geq 1$ послідовність випадкових величин, що мають нормальний розподіл, тобто $\varepsilon_t \sim N(0,1)$, σ_t позначають волатильність.

Тоді модель ARCH(p) являє собою умовну дисперсію як лінійну функцію квадратів попередніх значень величини, що спостережується, і записується наступним чином:

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^p \alpha_i y_{t-i}^2, \quad (2.27)$$

де p – максимальний порядок АРУГ,

σ_t^2 – умовна дисперсія,

y_{t-i} – попередні значення,

α_0, α_i – параметри моделі, і $\alpha_0 > 0, \alpha_i \geq 0, i = 1, \dots, p$, що пов'язане з тим, що дисперсія завжди повинна бути додатньою, проте при великій кількості лагів ця умова може бути порушена.

Модель АРУГ(1) $\sigma_t^2 = \alpha_0 + \alpha_1 y_{t-1}^2$ описує наступне твердження: якщо в деякий момент $t-1$, абсолютна величина похибки y_t має достатньо велике значення, це призводить до зростання умовної дисперсії у наступний момент t .

Ввівши позначення $\theta_t = y_t^2 - \sigma_t^2$, запишемо модель ARCH(p):

$$y_t^2 = \alpha_0 + \sum_{i=1}^p \alpha_i y_{t-i}^2 + \theta_t. \quad (2.28)$$

Оскільки y_{t-i} та v_t є незалежними величинами, можна побачити, що модель відповідає моделі AR(p) для квадратів залишків.

Це означає, що процес є слабо стаціонарним тільки тоді, коли сума додатніх параметрів авторегресії менша за 1. Безумовна дисперсія y_t має вигляд

$$E(\sigma^2) = \alpha_0 + \alpha_1 E(y_{t-1}^2) + \dots + \alpha_p E(y_{t-p}^2), \quad (2.29)$$

звідки $\sigma^2 = \frac{\alpha_0}{1 - \sum_{i=1}^p \alpha_i}$ за умови $0 \leq \alpha_1 + \dots + \alpha_p < 1$

Варто наголосити, що безумовна дисперсія не залежить від моменту часу t , тому має місце умовна гетероскедастичність.

Модель АРУГ не передбачає те, що процес y_t є нестаціонарним, а навпаки - те, що значення $y_t^2, y_{t-1}^2, \dots, y_{t-p}^2$ є корельованими.

Для перевірки гіпотези наявності АРУГ ефектів використовують LM-тест, запропонований Енгелем. Порядок дій наступний:

1. Нехай маємо процес x_t . Формування ряду залишків: $e_t = x_t - \hat{x}_t$.
2. Оцінюється наступна модель регресії:

$$e_t^2 = \alpha_0 + \sum_{i=1}^p \alpha_i e_{t-i}^2 + z_t. \quad (2.30)$$

3. Висувається гіпотеза про відсутність АРУГ ефекта:

$H_0: \alpha_1 = \dots = \alpha_p = 0$ проти альтернативної гіпотези;

H_1 : хоча б один з коефіцієнтів $\alpha_i \neq 0$.

4. Гіпотеза перевіряється за допомогою статистичного тесту Фішера (у випадках, коли розмір вибірки достатньо великий, використовують тест χ^2), де $\chi^2_{\text{факт}}$ порівнюється із табличним значенням.

Якщо гіпотеза H_0 виконується, отримана статистика $\chi^2_{\text{факт}}$ розподілена за законом χ^2 із p степенями свободи.

Оцінка коефіцієнтів АРУГ моделі.

Факт наявності АРУГ помилок в регресії або моделі авторегресії не впливає на застосування оцінки за допомогою МНК. Проте існують більш ефективні методи оцінювання. Якщо випадкові похибки розподілені за нормальним законом, має місце застосування методу максимальної правдоподібності.

При оцінюванні АРУГ моделей функція правдоподібності виглядає так:

$$\ln L = -\frac{1}{2} \sum_{t=1}^T \left(2 \ln \sigma_t + \frac{\varepsilon_t^2}{2\sigma_t^2} \right). \quad (2.31)$$

Важливою є можливість прогнозування умовної дисперсії за допомогою АРУГ моделі.

Запишемо альтернативну модель з використанням середнього значення:

$$\sigma_t^2 - \sigma^2 = \alpha_1 (y_{t-1}^2 - \sigma^2) + \dots + \alpha_p (y_{t-p}^2 - \sigma^2). \quad (2.32)$$

Тоді можна отримати прогноз на один крок уперед:

$$\sigma_{t+1|t}^2 = \sigma^2 + \alpha_1(y_t^2 - \sigma^2) + \dots + \alpha_p(y_{t-p+1}^2 - \sigma^2), \quad (2.33)$$

або на k кроків уперед:

$$\sigma_{t+k|t}^2 = \sigma^2 + \alpha_1(\sigma_{t+k-1|t}^2 - \sigma^2) + \dots + \alpha_p(\sigma_{t+k-p|t}^2 - \sigma^2). \quad (2.34)$$

Оскільки $\alpha_1 + \dots + \alpha_p < 1$, то при $k \rightarrow \infty$ прогноз k кроків уперед прямує до безумовної дисперсії, і як наслідок, не має практичного сенсу.

УАРУГ моделі.

Узагальнена авторегресійна умовно гетероскедастична модель (УАРУГ) (Generalized ARCH – GARCH модель) була запропонована Тімом Боллерслевом для моделювання волатильності, що залежить від часу. Вона включає значення попередніх умовних дисперсій, завдяки чому замість великих значень p моделі АРУГ(p) використовують невеликі значення p та q .

Модель УАРУГ визначає умовну дисперсію як лінійну комбінацію p попередніх квадратів залишків і q лагів попередніх значень умовної дисперсії:

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^p \alpha_i y_{t-i}^2 + \sum_{j=1}^q \beta_j \sigma_{t-j}^2, \quad (2.35)$$

де $\alpha_0, \alpha_i, \beta_j$ – параметри моделі, і $\alpha_0 > 0, \alpha_i \geq 0, \beta_j \geq 0, i = 1, \dots, p, j = 1, \dots, q$.

Модель УАРУГ(p, q) може бути записана так:

$$\sigma_t^2 = \alpha_0 + \alpha(L)y_{t-1}^2 + \beta(L)\sigma_{t-1}^2, \quad (2.36)$$

де $\alpha(L)$, $\beta(L)$ - поліноми оператора зсуву.

На практиці специфікація УАРУГ(1,1) працює дуже добре, її можна записати наступним чином:

$$\sigma_t^2 = \alpha_0 + \alpha y_{t-1}^2 + \beta \sigma_{t-1}^2, \quad (2.37)$$

де наявні лише три невідомі параметри.

Для невід'ємності σ_t^2 необхідно, щоб параметри α_0, α, β були також невід'ємними.

Щоб оцінити модель УАРУГ(p,q) за допомогою методу максимальної правдоподібності, необхідно рекурсивно розрахувати ряд $\{\sigma_t^2\}_{t=0}^{\infty}$, починаючи з нульового члена та припускаючи, що значення залишків та дисперсій $\{\sigma_{-1}^2, \dots, \sigma_{-q}^2, y_{-1}^2, \dots, y_{-p}^2\}$ є довільними.

Випадковий процес y_t моделі УАРУГ(p,q) із невід'ємними параметрами $\alpha_0, \alpha_i, \beta_j$ є стаціонарним тоді і тільки тоді, коли $\alpha + \beta < 1$ або $\sum_{i=1}^p \alpha_i + \sum_{j=1}^q \beta_j < 1$.

Тоді безумовна дисперсія виглядає наступним чином:

$$\sigma^2 = \frac{\alpha_0}{1 - (\alpha_1 + \dots + \alpha_p + \beta_1 + \dots + \beta_q)}. \quad (2.38)$$

Прогноз на k кроків уперед обчислюється наступним чином:

$$\sigma_{t+k|t}^2 = \sigma^2 + (\alpha + \beta)^k (\sigma_t^2 - \sigma^2). \quad (2.39)$$

Отже, прогнози УАРУГ(1,1) збігаються до безумовної дисперсії зі швидкістю $\alpha + \beta$.

Аналіз фінансових часових рядів дозволяє зробити висновок про те, що для поведінки фінансових параметрів характерна від'ємна кореляція y_{t-1} та σ_t , тобто $cov(y_{t-1}, \sigma_t) < 0$, а безумовна дисперсія залежить від розміру залишків, а не їх знаку. Тобто, наприклад, при збільшенні цін активів відбувається ріст дохідності, а волатильність зменшується, і навпаки, при зниженні цін та дохідності, волатильність зростає. Цей феномен отримав назву «ефект важеля» або «левередж-ефект».

Називаючись також ефектом асиметрії, цей феномен не може бути пояснений за допомогою АРУГ та УАРУГ моделей, оскільки волатильність σ_t^2 залежить від квадратів попередніх величин y_{t-1}^2 , а отже, не чутлива до їх знаків.

Завдяки цьому поширення набули асиметричні моделі.

Запропонована Д. Нільсоном експоненційна УАРУГ модель (exponential GARCH – EGARCH або ЕУАРУГ) враховує ефект асиметрії та описує часову залежність волатильності за допомогою авторегресійної залежності логарифма. Вона має наступний вигляд:

$$\log \sigma_t^2 = \alpha_0 + \beta \log \sigma_{t-1}^2 + \gamma \frac{y_{t-1}}{\sigma_{t-1}} + \alpha \frac{|y_{t-1}|}{\sigma_{t-1}}, \quad (2.40)$$

$$\log \sigma_t^2 = \alpha_0 + \beta \log \sigma_{t-1}^2 + \frac{y_{t-1}}{\sigma_{t-1}} (\gamma + \alpha), y_{t-1} > 0, \quad (2.41)$$

$$\log \sigma_t^2 = \alpha_0 + \beta \log \sigma_{t-1}^2 + \frac{y_{t-1}}{\sigma_{t-1}} (\gamma - \alpha), y_{t-1} < 0,$$

де α, β, γ – константи.

Оскільки модель ЕУАРУГ включає доданок $\frac{y_{t-1}}{\sigma_{t-1}}$, вона є асиметричною.

Якщо $\gamma < 0$, від’ємні збурення спричиняють більшу волатильність, ніж додатні.

Логарифмічне перетворення гарантує те, що дисперсії ніколи не будуть від’ємними, і умова $\alpha + \gamma > 0$ не є обов’язковою.

2.3 Моделі стохастичної волатильності

У моделях умовної гетероскедастичності наявне лише одне джерело випадковості, що задається нормально розподіленою послідовністю випадкових величин, тобто умовна дисперсія залежить від попередніх значень. Альтернативний спосіб побудови моделі полягає в тому, щоб волатильність σ процесу була окремим стохастичним процесом. Ця ідея лежить в основі моделей зі стохастичною волатильністю (або МСВ).

Для МСВ характерна наявність двох джерел випадковості: $\varepsilon = \varepsilon_t$ та $v = v_t$, що у найпростішому випадку є незалежними нормально розподіленими послідовностями випадкових величин, що визначають поведінку процесу $y = y_t$.

Серед перших моделей стохастичної волатильності можна назвати модель, що була запропонована Тейлором і представляла собою процес АР(1):

$$\begin{cases} y_t = \mu \sigma_t \varepsilon_t \\ \ln \sigma_t^2 = h_t = \phi \ln \sigma_{t-1}^2 + v_t \end{cases} \quad (2.42)$$

де μ — деяка константа, введення якої в модель дозволяє позбутися вільного члену у рівнянні волатильності, причому $\mu > 0$,

φ — параметр авторегресії, що визначає пам'ять волатильності.

Логарифм квадрата волатильності $\ln \sigma_t^2$ є новим випадковим процесом. Так як v_t — нормально розподілений процес, h_t є стандартною гаусівською авторегресією, і модель називають лог-нормальною.

За структурою ця модель є простішою за АРУГ, проте для неї характерна складність побудови оцінки. Для неї неможливо в явному вигляді записати функцію правдоподібності. Алгоритми оцінювання параметрів моделі стохастичної волатильності базуються на методі Монте-Карло для інтегрування багатовимірної щільності.

2.4 Моделі на основі нейронних мереж

Нейронна мережа — це математична модель, а також її програмне втілення, що побудована за принципом діяльності нервових клітин живих організмів - нейронів та являє собою сукупність згрупованих між собою простих процесорів.

При моделюванні кожного штучного нейрону мається на меті, що він являє собою елемент, що виконує достатньо прості функції — обробляє сигнали, що надходять до нього через декілька вхідних каналів (від інших процесорів або з навколишнього середовища). Об'єднуючись у велику мережу, нейрони здатні виконувати досить складні та громіздкі завдання та встановлювати взаємозв'язки між окремими даними.

Подібно до біологічного, штучний нейрон складається із синапсів, що забезпечують зв'язок вхідних сигналів із ядром, ядра, завданням якого є обробка

вхідних сигналів та аксона, що зв'язує даний нейрон із нейронами наступного шару. Кожний синапс має вагу, що визначає залежність вхідного сигналу та стану нейрона.

Математична модель нейрона:

$$y = f\left(\sum_{i=1}^n w_i x^i + b\right), \quad (2.43)$$

де y – вихідний сигнал нейрона,

f – функція активації (нелінійне перетворення),

n – кількість входів,

w_i – вага синапса,

x^i – вхідний сигнал,

b – зовнішнє джерело (значення зсуву).

Найчастіше в якості активаційної функції f виступають:

– сигмоїда: $y = \frac{1}{1 + e^{-\alpha x}}$;

– порогова бінарна функція: $f(S) = \begin{cases} 0, S \leq S_0 \\ 1, S > S_0 \end{cases}$,

при цьому нейрон залишається неактивним до тих пір, поки вхідне значення не стане рівним S_0 .

- сигнум: $f(S) = \begin{cases} -1, S < S_0 \\ 0, S = S_0 \\ 1, S > S_0 \end{cases}$;

- лінійна функція $y = kS$, де k – тангенс кута нахилу прямої.

Нейромережі являють собою інструмент, що дозволяє моделювати різноманітні залежності, такі як лінійні, нелінійні, узагальнені лінійні моделі. Здатність нейронних мереж до визначення та узагальнення взаємозалежностей між вхідними та вихідними даними дозволяє здійснювати прогнозування за допомогою НМ, наприклад, курсу валют, динаміки ринків, визначення курсу акцій підприємств на біржі.

Після процедури навчання НМ може прогнозувати майбутнє значення деякої послідовності факторів на основі їхніх попередніх значень. Для прогнозування майбутніх значень необхідно підготувати дані для навчання та перевірки функціонування НМ, обрати основні характеристики та алгоритми навчання нейромережі. На рівень деталізації даних впливають наступні чинники: точність та доступність даних, вартість аналізу і рівень переваги отриманих результатів.

Однією із задач, що розв'язуються за допомогою нейронних мереж, є задача прогнозування часових рядів. На практиці найчастіше поширений випадок прогнозування на один крок уперед на основі N попередніх значень.

Схему розв'язання задачі прогнозування вихідного часового ряду X можна представити у вигляді послідовності етапів:

1. Обрання глибини занурення – параметр τ ;
2. Поділ часового ряду на деяку кількість навчальних прикладів для формування навчальної вибірки.
3. Визначення структури нейронної мережі з фіксованою кількістю входів, рівною N .
4. Навчання нейромережі на основі навчальної вибірки.

Для розв'язання задачі прогнозування, у випадках, коли кожному вектору вхідних параметрів відповідає значення прогнозованого вектора, що представлений вихідними параметрами НМ, найчастіше здійснюють за допомогою підходу апроксимації функції. У результаті навчання параметри мережі зводяться до вигляду деякої функції, що представлена вхідними та вихідними параметрами навчальної вибірки.

На прогножуючу систему впливає точність прогнозу. Похибка прогнозу залежить від системи прогнозу, що використовується: прогноз тим точніший, чим більша кількість ресурсів задіяна при складанні системи. Точність прогнозу характеризується похибкою прогнозу.

Авторегресійні нейронні мережі (Neural Networks Autoregression, NNAR).

У випадку застосування авторегресійних нейронних мереж вхідними сигналами виступають лагові змінні часових рядів, вихідними – спрогнозовані значення. Однією з найголовніших відмінностей методів NNAR та ARIMA є те, що NNAR не накладає додаткових обмежень на параметри для забезпечення стаціонарності.

Модель NNAR(p, P, k)_m включає вхідні сигнали $\{y_{t-1}, y_{t-2}, \dots, y_{t-p}, y_{t-m}, \dots, y_{t-pm}\}$ та k нейронів на прихованому шарі. Структура моделі NNAR(p, P, k)_m зображена на рисунку 2.2:

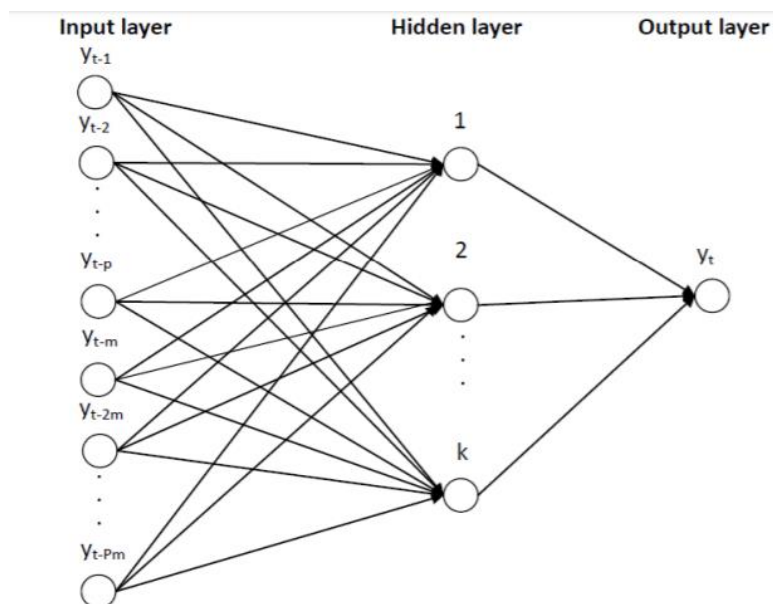


Рисунок 2.2 – Структура моделі NNAR(p, P, k)_m

За відсутності прихованого шару модель NNAR(p, P, 0)_m є еквівалентною моделі ARIMA(p,0,0)(P,0,0)_m.

Для операції прогнозування використовується ітеративний метод, тобто прогнозування на один крок уперед здійснюється на історичних даних. Здійснення операції прогнозування на два кроки уперед поряд із історичними даними потребує значення прогнозу, отриманого на попередньому кроці.

2.5 Моделі на основі методу групового урахування аргументів

Метод групового урахування аргументів (МГУА) використовується для аналізу багатопараметричних даних, моделювання та прогнозування систем, розпізнавання образів тощо, дозволяє визначати взаємозв'язки між даними, збільшувати точність існуючих алгоритмів.

МГУА є комплексом декількох алгоритмів, а саме: ймовірнісних, алгоритмів кластеризації, об'єднання аналогів в єдину систему. Метод базується на переборі моделей, що поступово ускладнюються, та виборі найкращої згідно з мінімумом зовнішнього критерія. В якості моделей застосовують лінійні, нелінійні, ймовірнісні функції та кластеризацію.

Основний результат МГУА полягає в тому, що за наявності зашумлених неточних даних з великою кількістю пропусків та при малих обсягах вибірок, мінімум критерію показує модель нефізичну, структура якої є простішою за структуру фізичної моделі, а точність – вища.

Алгоритми нейронних мереж типу МГУА визначають зв'язки між змінними, після чого обирається базова модель.

Перша модель МГУА була розроблена А. Г. Івахненко для моделювання многочленів вищих порядків та виглядала наступним чином:

$$y = a + \sum_{i=1}^m b_i x_i + \sum_{i=1}^m \sum_{j=1}^m c_{ij} x_i x_j + \sum_{i=1}^m \sum_{j=1}^m \sum_{k=1}^m d_{ijk} x_i x_j x_k + \dots \quad (2.44)$$

де m – кількість факторів, $(a, b_i, c_{ij}, d_{ijk})$ – вектор вагових коефіцієнтів,

y – залежна змінна, x_i, x_j – факторні змінні).

У загальному використовують модель наступного вигляду:

$$y = a + \sum_{i=1}^m b_i x_i + \sum_{i=1}^m \sum_{j=1}^m c_{ij} x_i x_j, \quad (2.45)$$

$$y = \omega_0 + \omega_1 x_i + \omega_2 x_j + \omega_3 x_i^2 + \omega_4 x_j^2 + \omega_5 x_i x_j, \quad (2.46)$$

де $(\omega_0, \omega_1, \omega_2, \omega_3, \omega_5)$ - вектор вагових коефіцієнтів.

Алгоритм МГУА враховує усі можливі комбінації двох випадкових величин та має наступні властивості:

1. Кожна комбінація подається на вхід нейрона.
2. Кількість нейронів на кожному рівні визначається кількістю вхідних змінних. Наприклад, якщо на вхід подається p незалежних змінних, і оскільки з них у якості вхідних включають дві, то кількість нейронів буде рівною $\frac{p(p-1)}{2}$.

Архітектура алгоритму МГУА для чотирьох факторних змінних зображена на рисунку 2.3:

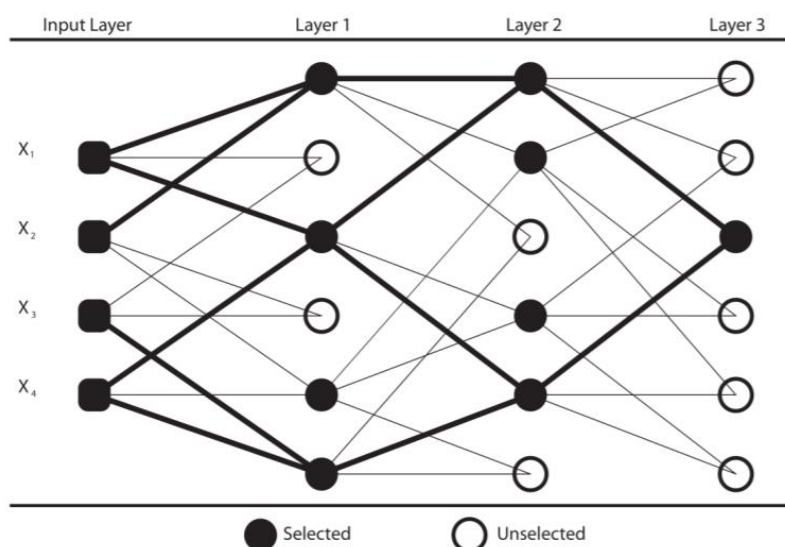


Рисунок 2.3 - Архітектура алгоритму МГУА

Вагові коефіцієнти моделі визначаються для кожного нейрона наступним чином:

$$\omega_k = (X_k' X_k + \lambda I_{6 \times 6})^{-1} X_k' y, \quad (2.47)$$

де X_k – коваріаційна матриця для k -го нейрону:

$$X_k = \begin{matrix} & 1 & x_{k,i,1} & x_{k,j,1} & x_{k,i,1}^2 & x_{k,j,1}^2 & x_{k,i,1}x_{k,j,1} \\ 1 & 1 & x_{k,i,2} & x_{k,j,2} & x_{k,i,2}^2 & x_{k,j,2}^2 & x_{k,i,2}x_{k,j,2} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & x_{k,i,t-p} & x_{k,j,t-p} & x_{k,i,t-p}^2 & x_{k,j,t-p}^2 & x_{k,i,t-p}x_{k,j,t-p} \end{matrix},$$

$$y = \begin{matrix} y_1 \\ y_2 \\ \dots \\ y_p \end{matrix}, \quad I_{6 \times 6} - \text{одинична матриця та}$$

$$\omega_k = [\omega_{0,k} \quad \omega_{1,k} \quad \omega_{2,k} \quad \omega_{3,k} \quad \omega_{4,k} \quad \omega_{5,k}]'.$$

3. Для обчислення коефіцієнтів регуляризації вхідні дані поділяють на дві вибірки: навчальну (70%) та тестову (30%). Алгоритм пошуку коефіцієнтів регуляризації складається з наступних кроків:

- уточнення можливих коефіцієнтів регуляризації:
 - $\lambda = 0, 0.01, 0.02, 0.08, \dots, 10.24$;
- для кожного коефіцієнта λ обчислюються вагові коефіцієнти на навчальній вибірці;
- після обчислення вагових коефіцієнтів, визначаються спрогнозовані значення для тестової вибірки та середні квадрати похибок для кожного коефіцієнта регуляризації;
- вибір коефіцієнта регуляризації, що дає найменший середній квадрат похибок.

4. У якості зовнішнього критерію оцінки якості моделі виступає прогнозний середній квадрат похибок

$$PMSE = \frac{1}{t-p} \sum_{i=1}^{t-p} [\hat{z}_i - z_i]^2, \quad (2.48)$$

де \hat{z}_i – спрогнозовані значення, z_i – реальні значення.

5. Якщо операції виконані не для останнього рівня нейромережі, обирають p вихідних значень із найменшим значенням критерію $PMSE$ та повертаються до кроку 3.

2.6 Критерії адекватності моделей

1. Коефіцієнт детермінації R^2 вказує на якість підбору регресійної моделі до спостережуваного значення, що обчислюється за формулою:

$$R^2 = \frac{var(\hat{y})}{var(y)} = 1 - \frac{SSE}{SST} = \frac{SSR}{SST}, \quad (2.49)$$

де $var(\hat{y})$ - дисперсія залежної змінної, оціненої за допомогою побудованої моделі;

$var(y)$ – дисперсія вимірів залежної змінної;

$SSE = \sum_{i=1}^N [y_i - \hat{y}_i]^2$ – сума квадратів похибок моделі;

$SST = \sum_{i=1}^N [y_i - \bar{y}]^2$ – загальна сума квадратів, де \bar{y} – середнє значення ;

$SSR = \sum_{i=1}^N [\hat{y}_i - \bar{y}]^2$ – загальна сума квадратів регресії;

Якщо коефіцієнт $R^2 = 0$, то регресія не дає якісний прогноз y_i порівняно із прогнозом $\hat{y}_i = \bar{y}$.

Випадок, коли коефіцієнт $R^2 = 1$ означає точність підбору, тобто дисперсії значень змінної, оцінених за моделлю, та дисперсій реальних змінних, збігаються.

Можна виділити наступні властивості R^2 :

- R^2 зростає при додаванні ще одного регресора;
- R^2 змінюється при будь-якому перетворенні залежної змінної, тому можна порівнювати значення R^2 тільки для моделей з однаковими змінними;
- якщо кількість змінних рівна кількості регресорів, можна отримати $R^2 = 1$, проте це не означає, що така залежність має економічний сенс.

Недоліком критерію R^2 є те, що його значення не буде зменшуватися при додаванні в модель регресорів, навіть якщо додаткові регресори не мають ніякої пояснюючої міцності. Загальним способом розв'язання у цьому випадку полягає в коригуванні оцінок дисперсії на степені свободи. Таким чином, вводимо скоригований коефіцієнт детермінації, що обчислюється за формулою:

$$R_{adj}^2 = 1 - \frac{SSR}{n-k} \frac{n-1}{SST}, \quad (2.50)$$

де n — кількість спостережень, k — кількість параметрів.

Цей показник передбачує певне покарання за включення у модель додаткових змінних, і тому автоматично не збільшується (або не завжди збільшується) при введенні додаткових регресорів. У деяких випадках R_{adj}^2 стає негативним. R_{adj}^2 строго менший за R^2 , крім випадків, коли модель складається із одного константного члена, або якщо обидва виміри не рівні нулю.

2. Статистика Дарбіна-Уотсона вказує на адекватність побудованої моделі та знаходиться за формулою:

$$DW = \frac{\sum_{t=2}^N [\varepsilon_t - \varepsilon_{t-1}]^2}{\sum_{t=2}^N \varepsilon_t^2}, \quad (2.51)$$

де ε_t – залишки, оцінені за МНК.

Запишемо $\sum_{t=2}^N [\varepsilon_t - \varepsilon_{t-1}]^2 = \sum_{t=1}^N \varepsilon_t^2 - \varepsilon_1^2 - 2 \sum_{t=1}^N \varepsilon_t \varepsilon_{t-1} + \sum_{t=2}^{N+1} \varepsilon_{t-1}^2 - \varepsilon_N^2$.

$$DW = 2 - 2p \frac{\sum_{t=2}^N \varepsilon_t^2}{\sum_{t=1}^N \varepsilon_t^2} - \frac{\varepsilon_N^2 + \varepsilon_1^2}{\sum_{t=1}^N \varepsilon_t^2} \approx 2 - 2p. \quad (2.52)$$

Отже, можна сказати, що область значень $DW \in [0; 4]$. За умови повної кореляції $DW = 2$.

3. Інформаційний критерій Акайке, що використовується для вибору найкращої моделі та обчислюється за формулою:

$$AIC = N \ln(\sum_{i=1}^N \varepsilon_i^2) + 2n = N \ln(SSR) + 2n \quad (2.53)$$

де n - кількість параметрів статистичної моделі.

Найкращою є та модель, у якої критерій Акайке найменший. Оскільки критерій Акайке враховує кількість параметрів, що оцінюються, та довжину вибірки, він є інформативним.

4. Критерій Байєса-Шварца є модифікацією критерія Акайке та рівен

$$BSC = N \ln(\sum_{i=1}^N \varepsilon_i^2) + n \ln(N), \quad (2.54)$$

де $n = p + q - 1$ – кількість параметрів моделі, що оцінюють за допомогою статистичних даних (p - кількість параметрів авторегресійної компоненти моделі; q - кількість параметрів ковзного середнього; 1 з'являється в тому разі, коли оцінюється зміщення a_0).

Особливість критерію полягає в тому, що зміна залежних змінних та збільшення кількості спостережуваних змінних призводить до збільшення критерію. Водночас зменшення значення статистики означає зменшення розмірності побудованої моделі. Зважаючи на асимптотичні властивості критерію, його краще застосовувати для великих вибірок даних ($N > 100$).

5. Статистика Стюдента або t – статистика застосовується для перевірки значущості коефіцієнта в статистичному сенсі і дорівнює

$$t = \frac{\hat{a} - a^0}{SE_{\hat{a}}}, \quad (2.55)$$

де \hat{a} - оцінка коефіцієнта моделі,

a^0 – початкова гіпотеза (нуль-гіпотеза) щодо цієї моделі,

$SE_{\hat{a}}$ – стандартна похибка оцінки.

В якості основної гіпотези висуваюють гіпотезу H_0 про несуттєву значимість параметра регресії або коефіцієнта кореляції ($a^0 = 0$). Альтернативною є гіпотеза про те, що параметр регресії або коефіцієнт кореляції не є рівним нулю ($a^0 \neq 0$).

Спостережуване значення t – критерію порівнюють з критичним значенням, що визначається за допомогою таблиці розподілу Стюдента, для цього використовують рівень значимості α та число степенів свободи (у випадку лінійної парної регресії рівне $n - 2$, де n – кількість спостережень.

Якщо спостережуване значення критерію за модулем більше за табличне ($|t| > |t_{кр}|$), то з ймовірністю $1 - \alpha$ коефіцієнт значимо відмінний від нуля. У протилежному випадку ($|t| < |t_{кр}|$), коефіцієнт незначимо відмінний від нуля, і основна гіпотеза приймається. Маємо висновок, що чим більшим є значення статистики для оцінки коефіцієнта, тим із більшою ймовірністю цей коефіцієнт є значимим.

б. F – статистика Фішера показує, наскільки добре модель пояснює загальну дисперсію залежної змінної. Для множинної регресії обчислюється наступним чином:

$$F = \frac{R^2}{1-R^2} \cdot \frac{N-p-1}{p}, \quad (2.56)$$

де p – кількість параметрів моделі без урахування константи,

N – кількість значень ряду.

Маємо, що якщо $R^2 \rightarrow 1$, то $F \rightarrow \infty$.

Нульовою гіпотезою є гіпотеза про те, що модель неадекватна ($H_0: a_0 = a_1 = \dots = a_p = 0$). Альтернативною є гіпотеза про те, що хоча б одне значення a_i є відмінним від нуля.

Значення $F_{кр}$ визначають із таблиць розподілу Фішера і якщо $F > F_{кр}$, нульова гіпотеза про неадекватність моделі в загальному сенсі відхиляється.

2.7 Критерії якості оцінок прогнозів

1. Сума квадратів похибок моделі:

$$\sum_{k=1}^N e^2(k) = \sum_{k=1}^N [\hat{y}(k) - y(k)]^2 \quad (2.57)$$

застосовується за наступним правилом : серед можливих моделей необхідно обирати ту, в якій сума квадратів похибок є меншою.

2. Середня похибка:

$$ME = \frac{1}{N} \sum_{i=1}^N y_i - \hat{y}_i. \quad (2.58)$$

3. Середня абсолютна похибка:

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|. \quad (2.59)$$

4. Середня відсоткова абсолютна похибка:

$$MAPE = \frac{1}{N} \sum_{i=1}^N \frac{|y_i - \hat{y}_i|}{|y_i|} \cdot 100\%. \quad (2.60)$$

5. Середня відсоткова похибка:

$$MPE = \frac{1}{N} \sum_{i=1}^N \frac{y_i - \hat{y}_i}{y_i} \cdot 100\%. \quad (2.61)$$

6. Середній квадрат похибок:

$$MSE = E((y_i - \hat{y}_i)^2) = \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{N}. \quad (2.62)$$

7. Коефіцієнт невідповідності Тейла:

$$U = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^N [y_i - \hat{y}_i]^2}}{\sqrt{\frac{1}{N} \sum_{i=1}^N y_i^2} + \sqrt{\frac{1}{N} \sum_{i=1}^N \hat{y}_i^2}}. \quad (2.63)$$

Величина коефіцієнта приймає значення від 0 до 1. Якщо $U=1$, прогнозовані і фактичні ряди не є корельованими, і модель не може дати якісного прогнозу. У разі, якщо $U=0$, прогнозовані значення ряду співпадають з реальними, і модель можна вважати ідеальною.

2.8 Висновки до розділу

Серед методів моделювання часових рядів розрізняють декілька окремих класів: застосування регресійних моделей, моделей методу Бокса-Дженкінса, на основі нейронних мереж.

Серед переваг регресійних моделей можна виділити їх гнучкість, простоту та швидкість отримання результату прогнозування. Важливим недоліком даного типу моделей є неможливість моделювання нелінійних процесів та явищ, а також складності із визначенням параметрів.

Моделі типу Бокса-Дженкінса можна назвати одними із найпопулярніших через одноманітність та простоту їх проектування, існує велика кількість наукових робіт, присвячених застосування авторегресійних моделей. Недоліком

авторегресійних моделей є їх лінійність та велика кількість параметрів, визначення яких є достатньо часо- та ресурсозатратною.

Основною перевагою моделей на основі нейронних мереж, порівняно із іншими, є їх нелінійність, а отже, здатність встановлювати нелінійні залежності між значеннями факторів, а також масштабованість та адаптивність. Складності із застосуванням моделей такого класу є обґрунтування вибору методу та ресурсозатратність їх навчання.

РОЗДІЛ 3

ПОБУДОВА МОДЕЛЕЙ ДЛЯ ОЦІНЮВАННЯ ТА ПРОГНОЗУВАННЯ ДОХОДНОСТІ КОМПАНІЇ INTEL CORPORATION

3.1 Аналіз звіту про фінансові результати

Звіт про фінансові результати (Income Statement або profit and loss statement, P&L) – це фінансовий звіт, що містить інформацію про фінансові результати діяльності компанії за певний період часу, її доходи та витрати. В ньому вказується дохід, що генерує компанія і які збитки вона понесла у результаті генерації цього доходу.

Базове рівняння, що лежить в основі звіту про фінансові результати:

$$Net\ Income = Revenue - Expense, \quad (3.1)$$

де Net Income – чистий прибуток,

Revenue – виручка або дохід від реалізації товарів та послуг певної категорії,

Expense – витрати цієї ж категорії.

Звіт про фінансові результати містить наступні показники:

1. Revenue – виручка, тобто обсяг товарів та послуг, помножений на їхню ціну, знаходиться на верхній позиції звіту і називається Top Line. Термін Net Revenue знаходиться із урахувань знижок, повернень та інших витяжок.

Revenue складається із Operating revenue та Non-operating evenue, тобто доходів, отриманих в результаті діяльності компанії, або ж іншої діяльності відповідно.

Показник Revenue має рости із кожним періодом, проявляючи помірну волатильність.

2. Net Income— чистий прибуток компанії, знаходиться внизу звіту про фінансові результати та називається Bottom Line.

3. Net Income також включає gains and losses, що є надходженнями та відтоками активів, що не пов'язані безпосередньо із діяльністю компанії. Наприклад, якщо компанія реалізує певну продукцію, вони стосуються доходу, а витрати підраховуються окремо. Проте якщо компанія продає надлишок капіталу, його вартість віднімається від Cost of sales, а остаточний результат записується як gain або loss.

4. Gross Profit – валовий прибуток, що визначається шляхом вирахування із обсягу чистого прибутку, отриманого від реалізації продукції, її собівартості.

5. Cost of sales – собівартість, що включає в себе усі витрати, необхідні для виробництва та реалізації продукції. Показник включає витрати на маркетинг, вартість робочої сили, матеріали, знижки від продажів та накладні витрати, доставку до користувачів.

6. Operating expenses (OPex)— операційні витрати, що включають плату за обладнання, інвентар, арендну плату, маркетинг, страхування, дослідження, реструктуризацію, розробку та впровадження нової продукції:

- Marketing, general and administrative (MG&A);
- Research and development (R&D);
- Restructuring and other charges;
- Amortization of acquisition-related intangibles— амортизаційні відрахування.

7. Operating income— операційний дохід, тобто виручка за відрахуванням собівартості, адміністративних та операційних затрат.

8. Interest Expense— витрати на виплату процентів, тобто проценти за облігаціями, позиками та кредитами, являє собою нараховані протягом облікового періоду проценти, а не суму процентів, виплачених за період.

9. Income before Taxes— прибуток до податкових відрахувань.

10. Earning per Share (EPS)— прибуток на акцію, тобто чистий прибуток компанії, розділений на середньозважену кількість випущених акцій протягом облікового періоду. Обчислюється за наступною формулою:

$$EPS = \frac{Net\ Income - Preferred\ dividends}{Weighted\ average\ number\ of\ shares\ outstanding}, \quad (3.2)$$

де Preferred dividends — дивіденти за привілейованими акціями.

EPS є найважливішим показником при визначенні ціни акції на фондовому ринку та інвестиційної ефективності компанії, а також компонентою, що використовується при розрахуванні співвідношення ціни та прибутку.

Чистий дохід компанії та його компоненти (наприклад, прибуток від операційної діяльності та прибуток до оподаткування) є критичними показниками для аналізу як власного капіталу, так і кредитоспроможності.

Фінансові коефіцієнти.

За допомогою фінансових коефіцієнтів можна визначити поточний фінансовий стан компанії, можливі проблеми та методи їх вирішення. Їх використовують:

- кредитори для оцінки кредитних ризиків;
- власники та фінансові менеджери для прийняття ефективних управлінських рішень та визначення напрямків і стратегій розвитку компанії;
- інвестори для прогнозування прибутків та дивідентів.

Існує більше 200 показників, що використовуються при фінансовому аналізі різних аспектів фінансового стану підприємства. Їх можна поділити на наступні групи:

- показники ліквідності, що визначають здатність покривати усі платежі та зобов'язання за рахунок поточних активів;

- показники прибутковості;
- показники фінансової стійкості;
- показники ринкової вартості.

1. *Gross margin*— коефіцієнт валового прибутку або маржинальності, тобто частина загального об'єму виручки компанії, що залишається після відрахування прямих витрат, пов'язаних із виробництвом та реалізацією товару або послуги, що визначається за формулою:

$$Gross\ margin = \frac{Gross\ Profit}{Revenue}. \quad (3.3)$$

Високий показник маржинальності вказує на високу прибутковість компанії, що в свою чергу, є індикатором її фінансової привабливості, адже різниці показників маржинальності різних підприємств відображають ефективність їх стратегій у прийнятті рішень.

Значення для компанії Intel Corp за четвертий квартальний період $Gross\ margin = \frac{11878}{20209.00} = 0.5877$ або 58.77%. Середнє значення показника маржинальності для індустрії напівпровідників у США за той же період становила 45.79%.

2. *EBIT Margin* — показник ЕБІТ-маржинальності, що використовується для визначення рентабельності та обчислюється за формулою:

$$EBIT\ Margin = \frac{EBIT}{Net\ Revenue} = \frac{Operating\ income}{Net\ Revenue}. \quad (3.4)$$

Показник ЕБІТ-маржинальності дозволяє оцінити фінансові результати компанії, не враховуючи впливу амортизаційної політики, розміру податків та структури капіталу (виплачених процентів), завдяки чому його застосовують для порівняння результатів декількох підприємств, що ведуть діяльність в одному

економічному секторі. Інвестори звертають увагу на цей показник із метою прогнозування повернення вкладених у діяльність компанії коштів.

3. Interest coverage ratio (або times interest earned, ICR) – коефіцієнт покриття процентів, що відображає здатність позичальника покривати відсотки за позиками та облігаціями. Застосовується кредиторами та інвесторами для визначення ризику кредитування компанії та її фінансової стійкості.

$$\text{Interest coverage ratio} = \frac{EBIT}{\text{Interest Expense}}. \quad (3.5)$$

Чим вищий цей показник, тим більше компанія є фінансово незалежною від кредиторів.

Якщо $ICR < 1$, це означає, що операційного прибутку компанії недостатньо для покриття платежів за відстоками.

Нормальним є ICR, що знаходиться у межах від 3 до 4, тобто дохід компанії забезпечує не лише можливість покривати заборговування, а й достатній резерв на випадок спаду ділової активності.

Занадто високий ICR вказує на обережний підхід до залучення позикових коштів, що може призвести до пониження рентабельності компанії.

Наприклад, ICR компанії Intel Corp за 2019 рік становив 40.8. Сума короткострокових та довгострокових боргів компанії станом на 28 грудня 2019 року становила \$29,001 млн, проте дохід компанії забезпечує покриття заюоргованості. Графік зміни за період з 2015 по 2019 роки проілюстровано на рисунку 3.1:

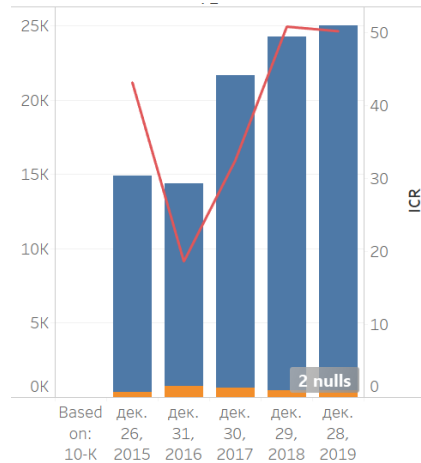


Рисунок 3.1 - Графік зміни ICR за 2015-2019 рр

4. Return on Assets (ROA) – показник рентабельності активів, характеризує ефективність використання активів для генерування виручки і обчислюється:

$$ROE = \frac{Net\ Income}{Average\ Total\ Assets}. \quad (3.6)$$

де Average Total Assets-величина активів підприємства.

Показник дозволяє охарактеризувати рівень професіоналізму менеджменту компанії, що впливає на розмір потенційного прибутку від інвестування.

5. Return on Equity (ROE)– показник рентабельності власного капіталу, тобто характеризує ефективність використання не усього капіталу підприємства, а тієї частини, що належить його власникам. Обчислюється за формулою:

$$ROE = \frac{Net\ Income}{Average\ Stockholder\ Equity}. \quad (3.7)$$

де Average Stockholder Equity - величина аукціонерного капіталу.

Показник ROE, що знаходиться у межах 0.15-0.2 вважається нормальним. Значення показника ROE Intel Corp за четвертий квартал 2019 року становив: $ROE = \frac{21,048}{77,504} = 0.2716$ або 27.16%. Для порівняння середнє значення ROE сектору напівпровідників, представником якої є Intel Corp, дорівнює 25.16%. Це свідчить про те, що компанія ефективно використовує свій капітал, а отже, є привабливою для інвесторів.

Графік зміни ROE за 2015-2019 роки проілюстровано на рисунку 3.2:

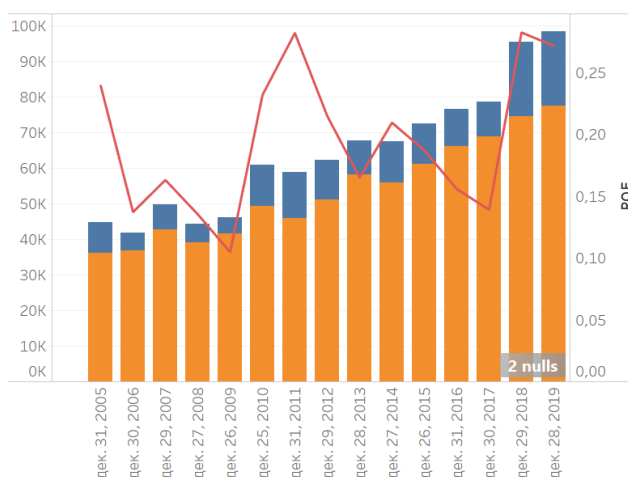


Рисунок 3.2 - Графік зміни ROE за 2015-2019 рр

6. Asset Turnover Ratio – показник оборотності активів, визначає ефективність використання виробничих ресурсів, розраховується за формулою:

$$Asset\ Turnover\ Ratio = \frac{Net\ Revenue}{Average\ Total\ Assets}. \quad (3.8)$$

7. Price-to-earnings ratio (P/E)- показник, що описує співвідношення між ціною акції та прибутком компанії, є важливим показником фундаментального аналізу акцій. В цілому (P/E) показує кількість років, за яку якції зможуть окупитися акції.

$$(P/E) = \frac{\text{Price per Share}}{\text{Earnings per Share}}. \quad (3.9)$$

Показник демонструє ступінь впевненості інвесторів у майбутньому компанії.

Станом на 20 березня 2020 року значення цього показника для Intel Corp було рівним 11.77.

3.2 Вибір програмного забезпечення для виконання обчислювальних експериментів

Інформаційна система для проведення фінансового аналізу та прогнозування показників компанії Intel Corp була розроблена за допомогою мови програмування для статистичної обробки даних R у програмному середовищі R Studio (рисунок 3.3).

Розроблена інформаційна система має наступні структурні елементи:

1. Стартова сторінка: завантаження та відображення бази даних, на основі яких проводиться моделювання та прогнозування фінансової стійкості компанії.
2. Крок 1. Відображення бази даних, у якій пропущені значення заповнені за допомогою методу найближчих сусідів.
3. Крок 2. Аналіз часового ряду:
 - побудова графіку часового ряду;
 - розклад вихідного ряду на трендову, сезонну та випадкову компоненти;
 - перевірка стаціонарності ряду за допомогою тестів Дікі-Фуллера (ADF тест), Філіпса-Перрона (PP тест), Квятковського – Філіпса – Шмідта – Шина (KPSS тест);
 - аналіз автокореляційної та часткової автокореляційної функцій;
 - логарифмування ряду, застосування першої та другої різниць для досягнення стаціонарності;

- можливість покращення моделі за допомогою згладжування вихідного ряду методом Хольта-Вінтерса або застосування фільтру Калмана.

4. Крок 3. Побудова ARIMA моделі, аналіз АКФ та АЧФ залишків моделі, аналіз залишків на наявність авторегресійної умовної гетероскедастичності (тести Бокса та Льюнга-Бокса), дослідження волатильності залишків та побудова ARCH/GARCH-моделі.

5. Крок 4. Прогнозування на основі ARIMA, ARIMA+ARCH моделей, методу групового урахування аргументів та нейронних мереж, оцінка критеріїв якості прогнозу.

6. Побудова моделі множинної регресії включає процедуру вибору факторів, що будуть включені у модель, аналіз корелограми регресорів, аналіз залишків побудованої моделі та побудову ARIMA моделі.

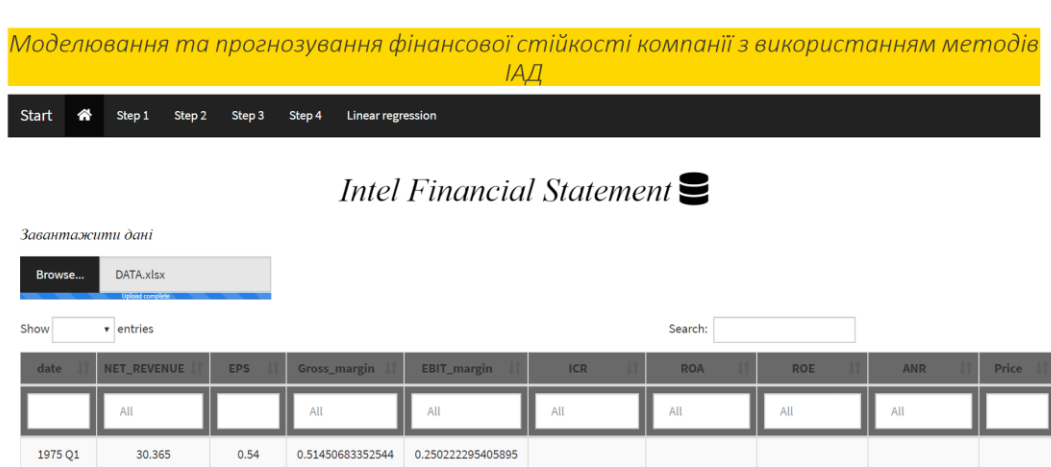


Рисунок 3.3 – Інтерфейс роботи програми.

3.3 Побудова моделей та аналіз отриманих результатів

Аналіз та прогнозування показника Net Revenue компанії Intel Corporation.

На основі результатів тестів на стаціонарність ряду (розширений тест Дікі-Фуллера та тест Філіпса-Перрона) можна зробити висновок про нестационарність ряду.

У лівій частині вікна інформаційної системи наявний ряд опцій, що дозволяють виконати перетворення часового ряду (логарифмування, диференціювання – перехід до попарних різностей сусідніх значень ряду та послідовне логарифмування із диференціюванням), метою яких є зведення ряду до стаціонарного (рисунок 3.4).

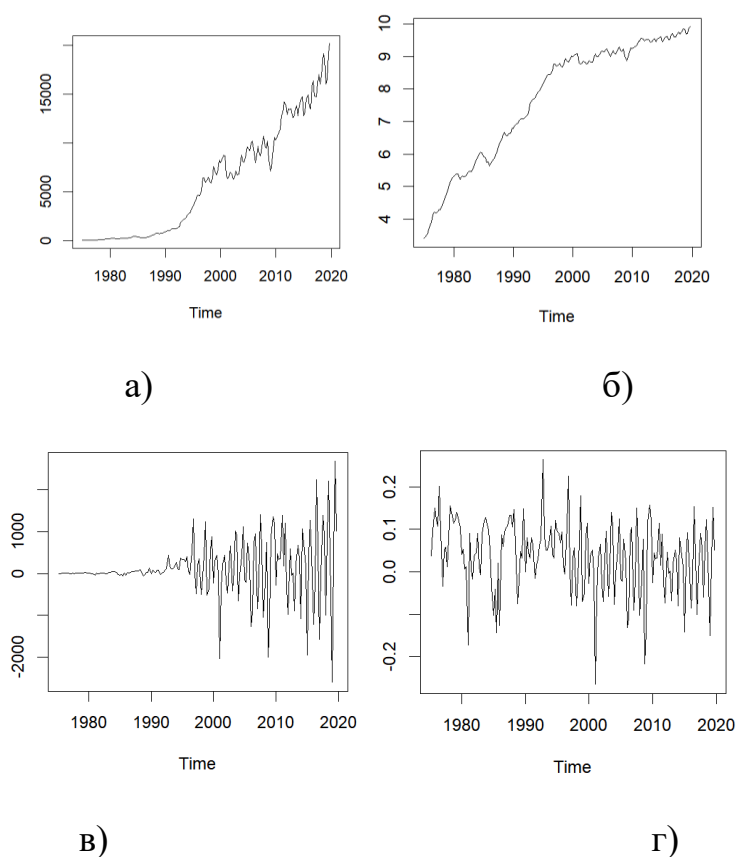


Рисунок 3.4 – Графік: а) Net Revenue, б) прологарифмований Net Revenue, в) продиференційований Net Revenue, г) прологарифмований та продиференційований Net Revenue

У випадках аналізу економічних та фінансових процесів найчастіше застосовують перетворення типу логарифмування, а потім взяття першої різниці. Це пов'язано із тим, що для таких часових рядів характерний експоненціальний ріст, який можна згладити за допомогою операції логарифмування, у той час як диференціювання стабілізує середнє значення ряду і дозволяє позбутися тренду та сезонності.

АКФ та ЧАКФ прологарифмованого та прологарифмованого та продиференційованого ряду виглядають наступним чином (рисунки 3.5, 3.6):

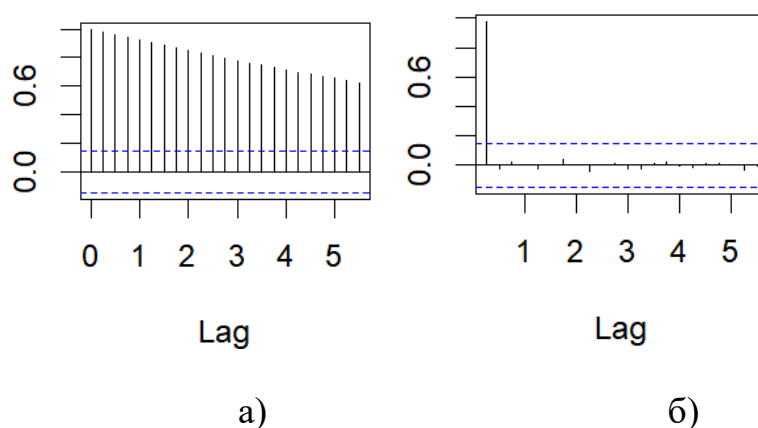


Рисунок 3.5 – а) АКФ, б) ЧАКФ прологарифмованого ряду

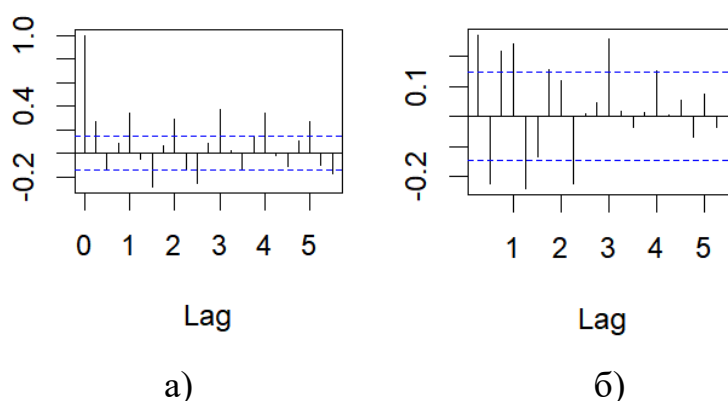


Рисунок 3.6 – а) АКФ, б) ЧАКФ прологарифмованого та продиференційованого ряду

Оскільки для часового ряду характерна відсутність нестационарності, ступені d та D сезонної моделі $SARIMA(p,d,q)(P,D,Q)[4]$ можуть бути рівними 1 та 2, 0, 1 та 2 відповідно, оптимальні значення показників p , q , P , Q необхідно обирати, використовуючи критерії Акайке та Байєса-Шварца.

Найкращою моделлю виявилася модель $SARIMA(0,2,1)(2,0,1)[4]$, для якої:
 $AIC = -406.5$ $AIC_c = -406.14$ $BIC = -390.7$.

Значення параметрів:

```
ma1 sar1 sar2 sma1
-0.9849 0.9005 0.0506 -0.7649
```

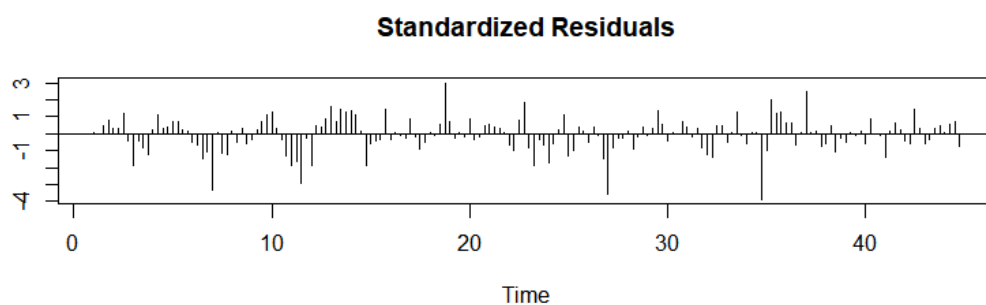


Рисунок 3.7 – Графік стандартизованих залишків моделі $SARIMA(0,2,1)(2,0,1)[4]$

За допомогою графіків АКФ та ЧАКФ залишків (рисунок 3.8) можна побачити, що значимих лагів немає, тому модель можна вважати адекватною та придатною для подальшого прогнозування.

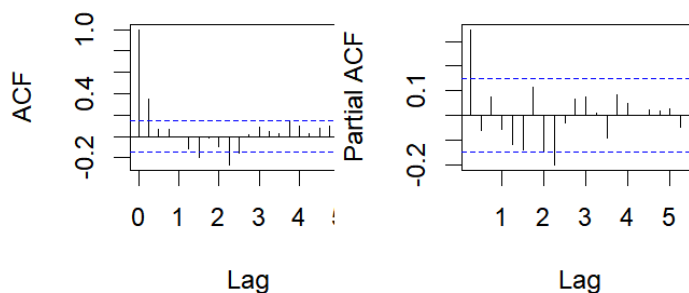


Рисунок 3.8 – а) АКФ, б) ЧАКФ залишків моделі $SARIMA(0,2,1)(2,0,1)[4]$

Наявність авторегресійної умовної гетероскедастичності у залишків моделі можна перевірити за допомогою тестів Бокса та Льюнга-Бокса, нульовою гіпотезою яких є випадковий розподіл даних. Значення критерію виявилося меншим за 0.05, отже, на рівні впевненості 95% можна стверджувати, що для залишків характерна взаємна корельованість.

Наступним кроком є побудова оптимальної ARCH/GARCH моделі для залишків моделі SARIMA (таблиця 3.1).

Таблиця 3.1 – Порівняння моделей ARCH/GARCH

Модель	<i>AIC</i>	<i>BIC</i>
ARCH(1)	-2.4089	-2.3729
ARCH(2)	-2.4106	-2.3566
GARCH(1,1)	-2.3968	-2.3427
GARCH(1,2)	-2.3855	-2.3134
GARCH(2,1)	-2.3854	-2.3134
GARCH(2,2)	-2.3741	-2.2841

Отже, найкращою моделлю виявилася модель ARCH(2).

Результати оцінювання моделі зображені на рисунку 3.9:

<i>Optimal Parameters</i>				
	<i>Estimate</i>	<i>Std. Error</i>	<i>t value</i>	<i>Pr(> t)</i>
<i>omega</i>	0.004598	0.000747	6.15443	0.00000
<i>alpha1</i>	0.018598	0.113494	0.16386	0.86984
<i>alpha2</i>	0.101433	0.088022	1.15236	0.24917

Рисунок 3.9 – Оцінені параметри для моделі ARCH(2)

Застосувавши моделі SARIMA(0,2,1)(2,0,1)[4], SARIMA(0,2,1)(2,0,1)[4] + ARCH(2), а також моделі на основі МГУА та нейронних мереж для прогнозування показника доходності компанії Intel Corporation за 2019 рік, отримано наступні результати проти реальних значень [9.684149; 9.711419; 9.862145; 9.913883] (таблиця 3.2):

Таблиця 3.2 – Порівняння якості оцінок прогнозу

Модель	Спрогнозовані значення	MSE	MAE	MAPE	THEILU
SARIMA(0,2,1)(2,0,1)[4]	9.773257 9.794229 9.870971 9.890275	0.0038582418844 5629	0.051088213761 9628	0.0052134318511 7684	0.0063174131215 9652
SARIMA(0,2,1)(2,0,1)[4] + ARCH(2)	9.773386 9.801191 9.883300 9.907986	0.0041261446108 7488	0.051515403938 8914	0.0052564064131 8171	0.0065268776487 7247
GMDH	9.812843 9.831248 9.836448 9.834217	0.0094820365994 2596	0.088471563896 7739	0.0090041978414 8538	0.0099072937240 7615
NNAR	9.794789 9.777724 9.792893 9.797546	0.0087419671146 2989	0.090633731248 7144	0.0092557203915 3677	0.0095496843136 0333
SARIMA(0,2,1)(2,0,1)[4] + NNAR	9.688160 9.745717 9.815720 9.792831	0.0045003173437 7418	0.051446396527 9743	0.0052560508115 4131	0.0068728885462 5651

Найкращі результати прогнозу отримані за допомогою звичайної моделі SARIMA(0,2,1)(2,0,1)[4], найгірші – моделі, побудованої на основі методу групового урахування аргументу.

У вікні “Step 2” інформаційної системи користувачеві надається можливість обрати покращення моделі – застосувати до часового ряду згладжування методом Хольта-Вінтерса або фільтр Калмана.

Виконавши аналогічні дії для отримання значень прогнозу показника доходності за 2019 рік, можна отримати результати (таблиця 3.3):

Таблиця 3.3 - Порівняння якості оцінок прогнозу, із використанням згладжування методом Хольта-Вінтерса та фільтра Калмана

Модель	Спрогнозовані значення	MSE	MAE	MAPE	THEILU
Хольт-Вінтерс SARIMA(1,2,2)(0,0,1)[4]	9.877616 9.904188 9.936306 9.962489	0.0188155197546 907	0.1353192074423 56	0.0136413855999 317	0.0138273069554 521
Хольт-Вінтерс SARIMA(1,2,2)(0,0,1)[4] + ARCH(4)	9.887128 9.925843 9.967427 10.001704	0.0264727660143 32	0.1606949477634 64	0.0161532708154 509	0.0163594176403 246
Хольт-Вінтерс GMDH	9.832966 9.831916 9.841332 9.842942	0.0040224985706 8211	0.0524583036496 886	0.0053342558609 7735	0.0064472196369 0779
Хольт-Вінтерс NNAR	9.838072 9.825942 9.825962 9.831290	0.0036732939236 2531	0.0508454318129 994	0.0051721848655 9634	0.0061653861176 0692
Хольт-Вінтерс SARIMA(1,2,2)(0,0,1)[4]+ NNAR	9.832237 9.911051 9.927265 9.902652	0.0140710914788 757	0.1084705542262 79	0.0109538852377 955	0.0119900150686 092
Калман SARIMA(2,2,3)(1,0,1)[4]	9.777421 9.782638 9.795230 9.813131	4.1993112123460 7e-05	0.0048505986443 6765	0.0004948616874 79984	0.0006617783153 07165

Продовження таблиці 3.3

Калман SARIMA(2,2,3)(1,0, 1)[4] + ARCH(4)	9.777534 9.783007 9.796365 9.815779	6.0370973603225 3e-05	0.0056755977628 4465	0.0005788327515 14059	0.0007933964477 01601
Калман GMDH	9.775345 9.770111 9.761720 9.750704	0.0009537604172 28717	0.0256834122757 543	0.0026319690491 8104	0.0031627931206 1836
Калман NNAR	9.779976 9.777926 9.772345 9.765046	0.0004515699061 36031	0.0163299661052 245	0.0016716063615 5147	0.0021741922401 4271
Калман SARIMA+NNAR	9.777094 9.782053 9.795078 9.811141	3.2157268714890 3e-05	0.0045429872300 593	0.0004636459437 41009	0.0005791578922 07037

Порівнявши отримані оцінки якості прогону, можна стверджувати, що найкращий прогноз дає модель SARIMA+NNAR, тобто застосування нейронних мереж для прогнозування залишків моделі SARIMA, де до часового ряду застосовано фільтр Калмана.

Аналіз та прогнозування показника Net Revenue компанії Intel Corporation на основі моделі множинної регресії.

У вікні “Linear regression” інформаційної системи користувачені запропоновано дослідити показник дохідності компанії із застосуванням моделі множинної регресії.

Першим кроком побудови моделі є визначення регресорів, що у неї входять. Існує два методи відбору факторів моделі: метод включення та метод виключення.

Спираючись на алгоритм методу виключення, необхідно відсіювати фактори до тих пір, поки модель не стане суттєво менш якісною. Таким чином, в результаті отримано, що у якості регресорів виступатимуть наступні показники: Gross Margin,

ICR, ROA, ROE, ANR, Price, для яких введемо позначення відповідно $X_1, X_2, X_3, X_4, X_5, X_6$ (рисунок 3.10).

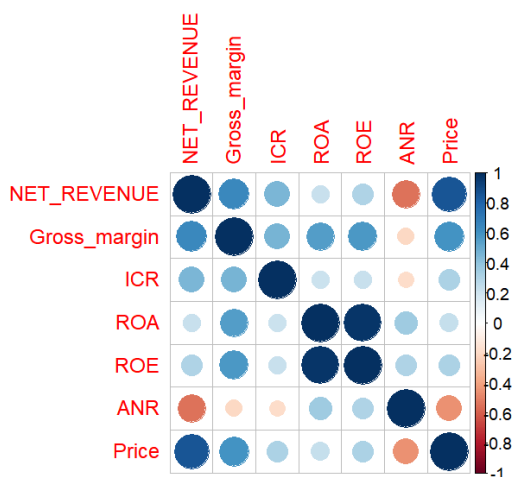


Рисунок 3.10 – Дослідження взаємної кореляції $Y, X_1, X_2, X_3, X_4, X_5, X_6$

Враховуючи отримані коефіцієнти, модель можна записати наступним чином:

$$Y = 8771.794X_1 + 10.165X_2 - 166.813X_3 + 136.720X_4 - 32.718X_5 + 206.680X_6 + 2755.078 + \varepsilon. \quad (3.10)$$

Оцінки адекватності моделі

[,1]	[,2]
"R squared"	"0.823688768537593"
"Adjusted R squared"	"0.817573928140053"
DW "Dw"	"0.675801362443861"
"AIC"	"3324.80957056574"
"BIC"	"3350.35322537286"

Оцінки якості прогнозу моделі

[,1]	[,2]
[1,] "MSE"	"941829.557998272"
[2,] "MAE"	"800.309020313319"
[3,] "MAPE"	"0.0440706910945356"
[4,] "Theilu"	"0.0533471196569216"

Рисунок 3.11 – Критерії якості моделі лінійної регресії

Аналіз АКФ та ЧАКФ залишків показав наявність значимих лагів на рисунку 3.12:

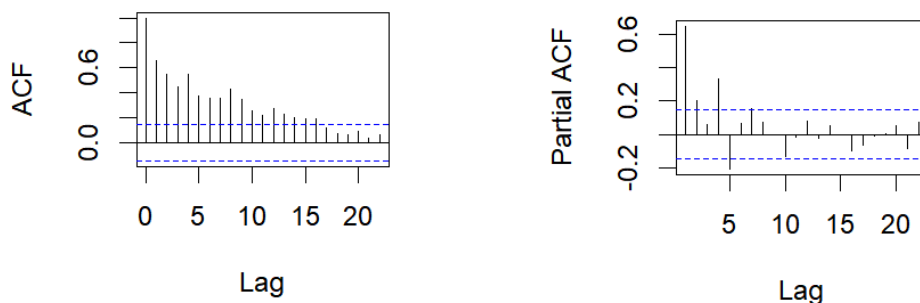


Рисунок 3.12 – а) АКФ, б) ЧАКФ залишків моделі лінійної регресії

Дослідження стаціонарності ряду за допомогою розширеного тесту Дікі-Фуллера та Філіпса-Перрона показали, що ряд є стаціонарним, проте результатом KPSS тесту є підтвердження гіпотези про наявність тренду.

Наступним кроком є побудова ARIMA моделі. Найкращою виявилася модель ARIMA(3,1,3), характеристики якої наведені на рисунку 3.13:

```

Coefficients:
            ar1            ar2            ar3            ma1
      -0.7962    -0.4537    -0.4484     0.3943
s.e.      0.1290     0.0917     0.0692     0.1361

sigma^2 estimated as 2783157:  log likelihood=-158
AIC=3170.95   AICc=3171.3   BIC=3186.89

```

Рисунок 3.13 – Оцінені параметри для моделі ARIMA(3,1,3)

Значення прогнозу показника Net Revenue на четвертий квартал 2019 року становить 20592.47, тоді як реальне значення – 20209. Отже, критерії якості прогнозу, побудованого на основі обраної моделі:

$$MAPE = 0.01897498, \text{ TheilU} = 0.01862163.$$

3.4 Висновки до розділу

Звіт про фінансові результати та розраховані на його основі фінансові коефіцієнти відображають результати діяльності компанії протягом певного звітнього періоду та використовують для оцінки доходності та здатності компанії покривати свої витрати.

У результаті проведення аналізу квартальних звітів про фінансові результати компанії Intel Corporation за 1975-2019 роки було розроблено інформаційну систему для моделювання та прогнозування доходів на основі наступних моделей:

- моделі множинної регресії;
- інтегрованої авторегресійної моделі – ковзного середнього;

- інтегрованої авторегресійної моделі – ковзного середнього з умовною гетероскедастичністю
- моделі на основі МГУА;
- моделі на основі нейронних мереж,

а також для порівняння результатів досліджень застосовано згладжування даних за методом Хольта-Вінтерса і фільтр Калмана. Перед проведенням подальших досліджень, вхідні дані підлягають попередній обробці, пропуски заповнено методом найближчих сусідів. Користувач має можливість самостійно обирати модель для побудови прогнозу, а також застосовувати описані вище способи маніпулювання над даними.

Усі моделі дають достатньо якісні оцінки прогнозів, тому їх можна застосовувати для прогнозування фінансових або економічних показників.

РОЗДІЛ 4

ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ ПРОГРАМНОГО ПРОДУКТУ

4.1 Постановка завдання проектування

У даному розділі роботи проводиться оцінка характеристик інформаційної системи, розробленої для моделювання та прогнозування економічних та фінансових показників, наприклад, доходності та розходів компаній, вартості акцій тощо. До основних технічних вимог програмного продукту можна віднести високу швидкість побудови математичних моделей, максимальну точність побудованих прогнозів, зручний функціонал та простоту у використанні. Програма розроблена та реалізована за допомогою програми для статистичної обробки даних R у програмному середовищі R Studio. Інтерфейс користувача спроектовано у інтегрованому середовищі Shiny. Експлуатація програмного продукту можлива на таких операційних системах, як Windows, MacOS, Ubuntu.

Проведення техніко-економічного аналізу продукту із метою оптимізації співвідношення його корисності та собівартості виконано із використанням підходів функціонально-вартісного аналізу продукту (ФВА).

4.1.1 Обґрунтування функцій програмного продукту

Головна функція F_0 реалізує основну мету розробки – аналіз процесів та показників за вхідними даними для подальшого моделювання та прогнозування. Кожна з основних функцій F_1, F_2, F_3, F_4 має декілька можливих варіантів реалізації.

Можна виділити наступні функції програмного продукту:

F_1 – вибір мови програмування: а) мова програмування R; б) мова програмування C++;

F_2 – збереження результатів роботи програми: а)лише виведення користувачу на екран; б)виведення користувачу на екран та збереження результатів у сховищі даних;

F_3 – розпізнавання вхідних даних: а)введення даних з файлу визначеного формату; б) написання програмного модулю для розпізнавання даних з використанням бібліотек комп'ютерного зору і машинного навчання;

F_4 – інтерфейс користувача: а) інтерфейс користувача, реалізований за допомогою програмного середовища Shiny; б) інтерфейс користувача, реалізований за допомогою інтерфейсу програмування додатків Windows Forms.

Морфологічна карта, на якій показано можливі варіанти реалізації основних функцій програмного продукту зображена на рисунку 4.1.

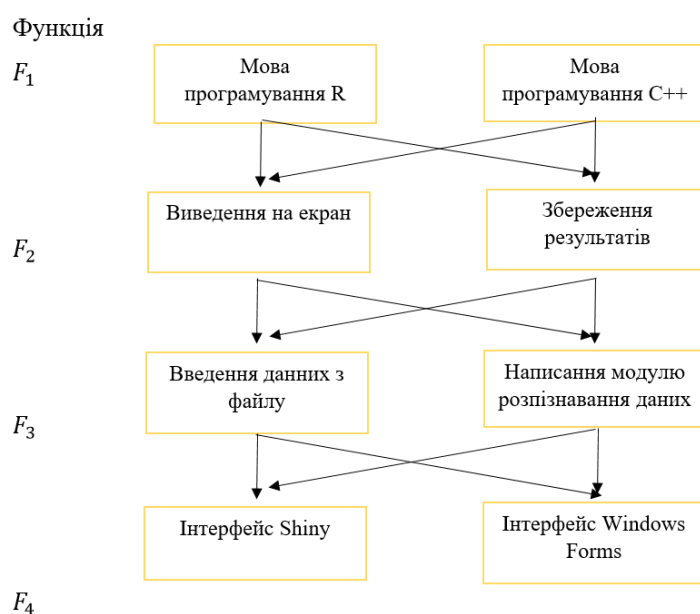


Рисунок 4.1 – Морфологічна карта

На основі даної морфологічної карти можна побудувати позитивно-негативну матрицю варіантів основних функцій (таблиця 4.1)

Таблиця 4.1 – Позитивно-негативна матриця

Основна функція	Варіант реалізації	Переваги	Недоліки
F_1	А	Кросплатформеність коду, великий масив функцій для роботи із даними	Вузькоспеціалізованість мови програмування, низька продуктивність коду
	Б	Високопродуктивний апарат мови програмування, загальнодоступність	Час написання коду.
F_2	А	Простота у використанні та реалізації	Недоступність результатів попередніх експериментів
	Б	Можливість відстеження і порівняння результатів	Складність реалізації
F_3	А	Простота у використанні та реалізації	Вимоги до формату даних, відсутність кросплатформеності
	Б	Широкий спектр можливостей, висока точність результатів експериментів	Складність та значні затрати часу, особливі вимоги до вхідних даних, необхідність встановлення додаткових бібліотек
F_4	А	Якісні пакети для візуалізації даних, достатньо малий час відклику	Необхідні своєчасні оновлення, додаткова інсталяція пакетів
	Б	Простота у використанні	Можливо використовувати лише у операційній системі Windows

На основі аналізу позитивно-негативної матриці можна дійти висновку, що при розробці програмного продукту варто відкинути деякі варіанти основних функцій через їх недостатню відповідність завданням, поставленим перед програмним продуктом.

Функція F_1 :

Оскільки робота проводиться із великими масивами вхідних даних, швидкість їх обробки має бути високою, тому варіант б) має бути відкинутий.

Функція F_2 :

Так як можливість збереження результатів проведених експериментів не є принциповою, можна розглядати обидва варіанти.

Функція F_3 :

Оскільки програмний продукт має на меті роботу із числовими даними, необхідність застосування методів машинного навчання відсутня, тому варіант б) можна не розглядати.

Функція F_4 :

Оскільки для роботи було обрано варіант а) функції F_1 , а також постає необхідність у якісній візуалізації даних, тому варіант б) має бути відкинтий.

Отже, розглянемо наступні варіанти використання програмного продукту:

$$F_1 \text{ а} - F_2 \text{ а} - F_3 \text{ а} - F_4 \text{ а},$$

$$F_1 \text{ а} - F_2 \text{ б} - F_3 \text{ а} - F_4 \text{ а}.$$

4.1.2 Обґрунтування системи параметрів програмного продукту

Для складення характеристики програмного продукту будемо використовувати систему параметрів $X1$ - $X5$, за умови, що функція F_1 залежить від двох параметрів- $X1$ та $X2$.

На основі вимог замовника та умов, що характеризують експлуатацію програмного продукту можна визначити гірші, середні і кращі значення параметрів.

Таблиця 4.2. – Основні параметри ПП

Назва параметра	Умовн е позна чення	Одиниц і виміру	Значення параметра		
			гірші	середні	кращі

Продовження таблиці 4.2

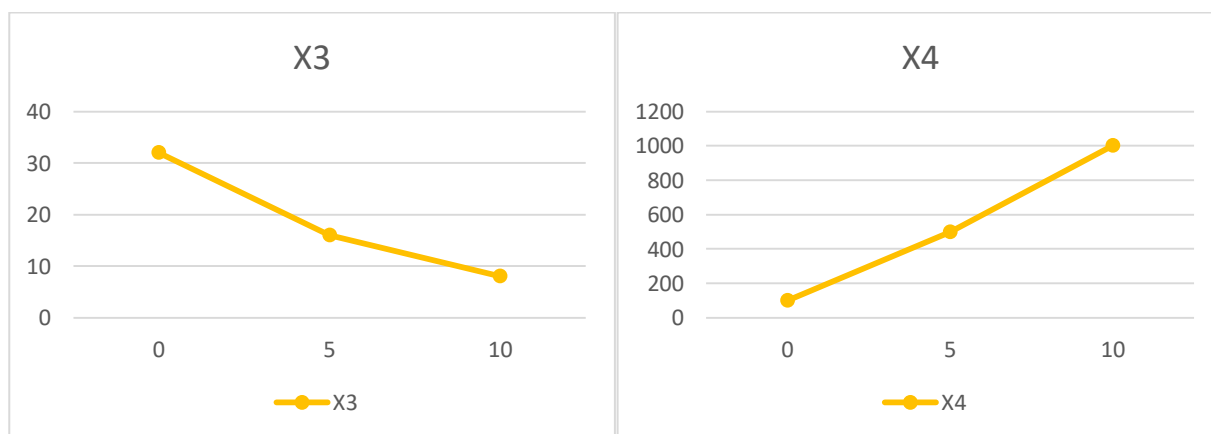
Час ознайомлення з мовою програмування	X1	год	15	10	6
Швидкість мови програмування	X2	Оп/мс	2000	11000	19000
Об'єм пам'яті для роботи програми	X3	Мб	32	16	8
Час обробки даних	X4	мс	1000	500	100
Потенційний об'єм програмного коду	X5	Кількість строк коду	1500	1000	500

За даними таблиці можна побудувати графічні характеристики параметрів (рисунок 4.2).



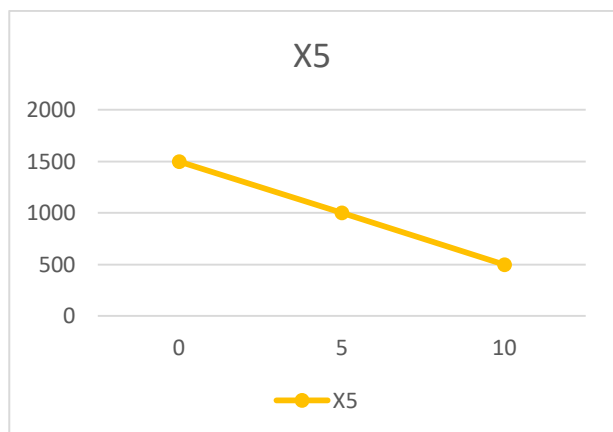
а)

б)



в)

г)



д)

Рисунок 4.2 – Графічні характеристики параметрів X1-X5

4.1.3 Аналіз експертного оцінювання параметрів

Вагомість кожного з параметрів $X1-X5$ визначається за допомогою методу попарного порівняння. Оцінки, проведені експертною комісією, що складається із 7 осіб, варюються від 1 до 5 і наведені у табл. 4.3. За найбільший ранг приймаємо оцінку 5, за найменший - 1.

Таблиця 4.3 – Результати ранжування параметрів

Назва параметра	Позначення параметра	Одиниці виміру	Ранг параметра за оцінкою експерта							Сума рангів в R_i	Відхилення Δ_i	Квадрат відхилення Δ_i^2
			1	2	3	4	5	6	7			
Час ознайомлення з мовою програмування	$X1$	год	1	2	2	1	1	2	2	11	-10	100

Продовження таблиці 4.3

Швидкодія мови програмування	X2	Оп/мс	3	3	3	4	3	4	4	24	3	9
Об'єм пам'яті для збереження даних	X3	Мб	2	1	1	2	2	1	1	10	-11	121
Час обробки даних	X4	мс	4	4	5	5	4	3	5	30	9	81
Потенційний об'єм програмного коду	X5	Кількість строк коду	5	5	4	3	5	5	3	30	9	81
	Разом		15	15	15	15	15	15	15	105	0	392

Таблиця 4.4 - Попарне оцінювання експертів

Параметри	Експерти							Кінцева оцінка	Числове значення
	1	2	3	4	5	6	7		
X1 і X2	<	<	<	<	<	<	<	<	0.5
X1 і X3	<	>	>	<	<	>	>	>	1.5
X1 і X4	<	<	<	<	<	<	<	<	0.5
X1 і X5	<	<	<	<	<	<	<	<	0.5
X2 і X3	>	>	>	>	>	>	>	>	1.5
X2 і X4	<	<	<	<	<	>	<	<	0.5
X2 і X5	<	<	<	>	<	<	>	<	0.5
X3 і X4	<	<	<	<	<	<	<	<	0.5
X3 і X5	<	<	<	<	<	<	<	<	0.5
X4 і X5	<	<	>	>	<	<	>	<	0.5

Визначимо коефіцієнт узгодженості

$$W = \frac{12S}{N^2(n^3-n)} = \frac{12 \cdot 392}{7^2(5^3-5)} = 0.8 > W_k = 0.67, \quad (4.1)$$

де N – число експертів,

n – кількість параметрів.

Оскільки значення даного коефіцієнта виявилося більшим за нормативне, ранжування можна вважати достовірним.

Таблиця 4.5 - Розрахунок вагомості параметрів

Параметри X_i	Параметри X_j					Перша ітерація		Друга ітерація		Третя ітерація	
	X1	X2	X3	X4	X5	b_i	K_{bi}	b_i^1	K_{bi}^1	b_i^2	K_{bi}^2
X1	1.0	0.5	1.5	0.5	0.5	4.0	0.16	17.5	0.152	80.25	0.153
X2	1.5	1.0	1.5	0.5	0.5	5.0	0.2	22	0.191	100	0.191
X3	0.5	0.5	1.0	0.5	0.5	3.0	0.12	14	0.122	64.5	0.122
X4	1.5	1.5	1.5	1.0	0.5	6.0	0.24	27.5	0.239	124.75	0.238
X5	1.5	1.5	1.5	1.5	1.0	7.0	0.28	34	0.296	155.5	0.296
Разом						25.0	1	115.0	1	525	1

Значення відносних оцінок вагомості, розрахованих на другій ітерації, відрізняються від попередніх значень менше, ніж на 5%.

Таблиця 4.6 - Розрахунок показників рівня якості варіантів реалізації основних функцій ПП

Основні функції	Варіант реалізації функції	Параметр	Абсолютне значення параметра	Бальна оцінка параметра	Коефіцієнт вагомості параметра	Коефіцієнт рівня якості
F_1	А	X1	6	10	0.153	1.53
		X2	19000	10	0.191	1.91
F_2	А	X3	8	10	0.122	0.122
	Б	X3	16	5	0.122	0.61

Продовження таблиці 4.6

F_3	A	X4	100	10	0.238	2.38
F_4	A	X5	800	6	0.296	1.776

Розрахуємо рівень якості кожного варіанту виконання кожної з основних функцій.

$$K_{K_1} = 1.53 + 1.91 + 1.776 + 2.38 + 1.22 = 8.816,$$

$$K_{K_2} = 1.53 + 1.91 + 1.776 + 2.38 + 0.61 = 8.206. \quad (4.2)$$

Отже, перший варіант за значенням коефіцієнта технічного рівня виявився кращим.

4.2 Економічний аналіз варіантів розробки програмного продукту

Для визначення вартості використання програмного продукту включає в себе розрахунок трудосмісткості.

Варіанти реалізації включають розв'язання наступних задач:

1. Розробка методів аналізу та попередньої обробки вхідних даних;
2. Підготовка проекту програмного продукту;
3. Розробка програмної оболонки.

Перший варіант має на меті відсутність ускладнень, другий варіант містить ускладнення, пов'язані із особливостями збереження даних (задача 4).

Задача 1 відноситься до групи В за ступенем новизни та до третьої групи складності. Для неї характерні: $T_p = 12$ людино-днів; поправочний коефіцієнт, що враховує вид нормативно-довідкової інформації $K_{\Pi} = 0.6$; поправочний коефіцієнт, що враховує складність вхідної та вихідної інформації $K_{ск} = 1$; коефіцієнт стандартного математичного забезпечення $K_{ст.м.} = 1.6$; коефіцієнт, що враховує використання стандартних модулів $K_{ст} = 0.8$.

Трудовісткість першої задачі дорівнює

$$T_1 = 12 * 0.6 * 1 * 1.6 * 0.8 = 9.216 \text{ людино-днів.} \quad (4.3)$$

Задача 2 відноситься до групи А за ступенем новизни та до першої групи складності, отже для неї $T_p = 90$ людино-днів, $K_{\Pi} = 1.7$, $K_{ск} = 1$, $K_{ст} = 0.8$.

$$T_2 = 90 * 1.7 * 1 * 0.8 = 122.4 \text{ людино-днів.} \quad (4.4)$$

Задача 3 відноситься до третьої групи складності та до групи Б за ступенем новизни, тоді $T_p = 19$ людино-днів, $K_{\Pi} = 0.9$, $K_{ст} = 0.8$.

$$T_3 = 19 * 0.9 * 0.8 = 13.68 \text{ людино-днів.} \quad (4.5)$$

Загальна трудовісткість виконання трьох задач (у випадку виконання першого варіанту) є сумою трудовісткості відповідних задач і дорівнює $T_1 = 145.296 \approx 145.3$ людино-днів.

У випадку другого варіанту завдання маємо задачу 4, що відноситься до групи Б за ступенем новизни та до третьої групи складності, для якої $T_p = 19$ людино-днів, $K_{\Pi} = 0.9$, $K_{ст} = 0.8$.

$$T_4 = 19 * 0.9 * 0.8 = 13.68 \text{ людино-днів.} \quad (4.6)$$

Тоді $T_{II} = 158.976 \approx 158.98$ людино-днів.

Вважаємо, що робочий місяць складається із 21 робочого дня, робочий день – з 8 робочих годин. В розробці програмного продукту задіяно одного аналітика даних (Data Analyst) із окладом $\$1200 \approx 32085$ грн.

Заробітня плата за годину обчислюється за формулою

$$C = \frac{M}{T_m \cdot t} = \frac{32085}{21 \cdot 8} = 190.98 \text{ грн.} \quad (4.7)$$

Тоді заробітня плата розробників рівна

$$\begin{aligned} C_I &= 190.98 \cdot 145.3 = 27749.39 \text{ грн.,} \\ C_{II} &= 190.98 \cdot 158.98 = 30356.27 \text{ грн.} \end{aligned} \quad (4.8)$$

Відрахування на соціальний внесок становить 22% від заробітньої плати:

$$\begin{aligned} C_{\text{відI}} &= 0.22 \cdot 27749.39 = 6104.87 \text{ грн.,} \\ C_{\text{відII}} &= 0.22 \cdot 30356.27 = 6678.38 \text{ грн.} \end{aligned} \quad (4.9)$$

Обчислимо витрати на оплату однієї машино-години. Оскільки комп'ютерна система обслуговує аналітика даних з окладом 32085 грн., з коефіцієнтом зайнятості $K_3 = 0.2$, для однієї машини маємо

$$C_{\Gamma} = 12 \cdot M \cdot K_3 = 12 \cdot 0.2 \cdot 32085 = 77004 \text{ грн.} \quad (4.10)$$

Враховуючи додаткову заробітню плату, маємо

$$C_{\text{зп}} = C_{\Gamma}(1 + K_3) = 77004(1 + 0.2) = 92404.8 \text{ грн.} \quad (4.11)$$

Відрахування на соціальне страхування складають

$$C_{\text{від}} = 0.22 * 92404.8 = 20329.06 \text{ грн.} \quad (4.12)$$

Обчислимо амортизаційні відрахування за амортизації у 25% та вартості ЕОМ 30000 грн за формулою:

$$C_A = K_{\text{ТМ}} * K_A * C_{\text{ПР}} = 1.15 * 0.25 * 20000 = 8625 \text{ грн.} \quad (4.13)$$

Обчислимо витрати на ремонт та профілактику ЕОМ:

$$C_P = K_{\text{ТМ}} * C_{\text{ПР}} * K_P = 1.15 * 0.05 * 30000 = 1725 \text{ грн.} \quad (4.14)$$

Обчислимо ефективний годинний фонд часу ПК:

$$\begin{aligned} T_{\text{еф}} &= (D_K - D_B - D_C - D_P) * t_3 * K_B = \\ &= (365 - 96 - 8 - 16) * 8 * 0.9 = 1764 \text{ год.} \end{aligned} \quad (4.15)$$

Витрати на оплату електроенергії знаходимо за формулою:

$$\begin{aligned} C_{\text{ЕЛ}} &= T_{\text{ЕФ}} * N_C * K_3 * C_{\text{ЕН}} = \\ &= 1764 * 0.2 * 2 * 1.46255 * 1.75 = 1805.96 \text{ грн.} \end{aligned} \quad (4.16)$$

Знаходимо накладні витрати:

$$C_H = C_{\text{ПР}} * 0.67 = 30000 * 0.67 = 20100 \text{ грн.} \quad (4.17)$$

Отже, експлуатаційні витрати становлять

$$\begin{aligned} C_{\text{екс}} &= 92404.8 + 20329.06 + 8625 + 1725 + 1805.96 + 20100 = \\ &= 144989.82 \text{ грн.} \end{aligned} \quad (4.18)$$

Собівартість однієї машино-години рівна

$$C_{\text{м-г}} = \frac{C_{\text{екс}}}{T_{\text{еф}}} = \frac{144989.82}{1764} = 82.19 \text{ грн/год.} \quad (4.19)$$

Усі роботи, задіяні на розробку та реалізацію програмного продукту виконуються за допомогою ЕОМ, тому витрати на оплату машинного часу знаходяться наступним чином:

$$C_{\text{мI}} = C_{\text{м-г}} T_{\text{I}} = 82.19 * 145.3 * 8 = 95537.66 \text{ грн.}, \quad (4.20)$$

$$C_{\text{мII}} = C_{\text{м-г}} T_{\text{II}} = 82.19 * 158.98 * 8 = 104532.53 \text{ грн.} \quad (4.21)$$

Накладні витрати рівні 67% витрат на оплату машинного часу:

$$C_{\text{нI}} = 0.67 * 95537.66 = 64010.23 \text{ грн.},$$

$$C_{\text{нII}} = 0.67 * 104532.53 = 70036.80 \text{ грн.} \quad (4.22)$$

Тоді загальна вартість розробки програмного продукту становить

$$C_{\text{ппI}} = 27749.39 + 6104.87 + 95537.66 + 64010.23 = 193402.15 \text{ грн},$$

$$C_{\text{ппII}} = 30356.27 + 6678.38 + 104532.53 + 70036.80 = 211603.98 \text{ грн.} \quad (4.23)$$

Розрахуємо коефіцієнт техніко-економічного рівня, скориставшись формулою:

$$K_{\text{тер1}} = \frac{K_{\text{к1}}}{C_{\text{ф1}}} = \frac{8.816}{193402.15} = 4.56e^{-5}, \quad (4.24)$$

$$K_{\text{тер2}} = \frac{K_{\text{к2}}}{C_{\text{ф2}}} = \frac{8.206}{211603.98} = 3.88e^{-5}. \quad (4.25)$$

Найбільш ефективним є перший варіант реалізації програмного продукту.

4.3 Висновки до розділу

У четвертому розділі роботи було проведено функціонально-вартісний аналіз програмного продукту, розробленого для моделювання та прогнозування економічних та фінансових показників. ФВА було проведено у два етапи.

У результаті проведення першого етапу – функціонального аналізу, було визначено чотири основні функції та визначено оптимальні варіанти їх реалізації, за допомогою яких реалізовано завдання, поставлені перед програмним продуктом.

У ході виконання другого етапу – вартісного аналізу, були розраховані основні витрати, виділені на створення програмного продукту, такі як витрати на оплату машинного часу та людських ресурсів, трудоемкість тощо.

Було отримано висновок, що найбільш оптимальним, із коефіцієнтом техніко-економічного рівня, рівним $K_{\text{тер}} = 4.17e^{-5}$, є варіант продукту, для якого характерні:

- мова реалізації – R;
- інтерфейс користувача – Shiny R;
- введення даних із файлу та виведення результатів роботи програми на екран.

ВИСНОВКИ

У даній роботі було розглянуто основні методи і моделі для задачі аналізу та прогнозування фінансових показників та процесів, а саме:

- моделі авторегресії, авторегресії – інтегрованого ковзного середнього;
- моделі авторегресії з умовною гетероседастичністю;
- моделі на основі методу групового урахування аргументу;
- моделі на основі нейронних мереж;
- моделі лінійної регресії.

Було досліджено основні критерії якості моделей та оцінки прогнозів. Було опрацьовано та реалізовано методи фільтрації та згладжування даних – Хольта-Вінтерса та фільтр Калмана, основним призначенням застосування яких було покращення обраних моделей.

Було розроблено інформаційну систему, за допомогою якої виконано моделювання та прогнозування фінансової діяльності компанії Intel Corporation. Даними для проведення експериментів слугували значення, розміщені у звітах про фінансові результати компанії за 1975-2019 рр, розміщених на її офіційній сторінці.

За результатами випробувань найкращою моделлю для прогнозування стала гібридна модель авторегресії – інтегрованого ковзного середнього та нейронної авторегресії.

Для подальших досліджень можна виділити наступні напрямки:

- реалізація заповнення пропущених значень за допомогою різних методів, наприклад, випадкового лісу або бустингу;
- реалізація додаткових методів моделювання та прогнозування, наприклад, на основі моделей стохастичної волатильності, динамічних байєсівських мереж тощо;
- модернізація інтерфейсу користувача із можливістю впорядковувати і зберігати отримані результати.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Вербик М. Путеводитель по современной эконометрике [Текст] / М. Вербик; пер. с англ. В.А. Банникова.; научн. ред. и пред. С.А. Айвазяна. - М.: Научная книга, 2008. – 616 с.
2. Магнус Я.Р. Эконометрика: Начальный курс [Текст] : учеб. / Я.Р. Магнус, П.К. Катышев, А.А. Песесецкий – 6-е изд., перед. и доп. – М.: Дело, 2004. – 576 с.
3. Трегуб А.В. Методика построения модели ARIMA для прогнозирования динамики временных рядов [Текст] / А.В. Трегуб, И.В. Трегуб. – Лесной вестник. – 2011. – №5. – С. 179-183.
4. Канторович Г.Г. Анализ временных рядов [Текст] / Г.Г. Канторович. – Экономический журнал ВШЭ. – 2002. – №2. – С. 251-273.
5. Саетова Л.Г. Основные модификации ARCH моделей [Текст] / Л.Г. Саетова. – Проблемы экономики и менеджмента. – 2014. – №6(34). – С. 61-63.
6. Истигичева А.В. Модели с авторегрессионной условной гетероскедастичностью [Электронный ресурс] / А.В. Истигичева, А.А. Мицель. – Режим доступа: <http://old.tusur.ru/filearchive/reports-magazine/2006-5/15-21.pdf>
7. Khalek. Md. Comparative Study of Wavelet-SARIMA and Wavelet- NNAR Models for Groundwater Level in Rajshahi District [Text] / Md. Abdul Khalek, Md. Ayub Ali. – IOSR Journal of Environmental Science, Toxicology and Food Technology. – July 2016. – P. 01-15.
8. Chatfield Chr. The Analysis of Time Series. An Introduction with R [Text] / Chr. Chatfield, H. Xing. – [7th ed.] – Taylor & Francis Group, LLC, 2019. – 414 p.
9. Dag O. GMDH: An R Package for Short Term Forecasting via GMDH-Type Neural Network Algorithms [Text] / O. Dag, C. Yozgatligil. – The R Journal. – Vol. 8/1. – Aug. 2016. – P. 379-386.

10. Кузнєцова Н.В. Методи і моделі аналізу, оцінювання та прогнозування ризиків у фінансових системах [Текст] : дис. ... д.т.н.: 01.05.04 / Кузнєцова Н.В. – Київ, 2018. – 415 с.
11. Бідюк П.І. Система підтримки прийняття рішень для аналізу даних [Текст] / П.І. Бідюк, Н.В. Кузнєцова, О.М. Терентьєв. – Наукові вісті НТУУ «КПІ». – 2011. – №1. – С. 48-61.
12. Shumway R.H. Time Series Analysis and Its Applications. With R examples [Text] / R.H. Shumway, D.S. Stoffer. – [3d ed.] - Springer Science+Business Media, LLC, 2011. – 202 p.
13. Операционный менеджмент [Текст] : уч. для вузов / С.Э. Пивоваров, И.А. Максимцев, И.Н. Рогова, Хутиева Е.С. – СПб: Питер, 2011. – 544 с.
14. Аджиева А.А. Использование фильтра Калмана в обработке данных хода напряженности электрического поля атмосферы [Электронный ресурс] / А.А. Аджиева, И.Х. Машуков, В.А. Шаповалов. - Инженерный вестник Дона. – 2017. – №2. – Режим доступа:
http://ivdon.ru/uploads/article/pdf/IVD_200_Adzhieva_Shapovalov_Mashukov.pdf_1a9c51afb0.pdf
15. С. Ю. Анисимов. Цифровые фильтры в SCADA [Электронный ресурс] / Анисимов С.Ю. – Режим доступа: http://www.pmg.org.ru/asutp/scada_filter.pdf
16. Кошечкин А.А. Новый метод восстановления пропущенных значений в наборе данных на примере иммуносигнатур [Электронный ресурс] / А.А. Кошечкин, В.С. Андрющенко, А.В. Замятин. – СТМ. – Том 12. - №2. – С.19-24.– Режим доступа: <https://cyberleninka.ru/article/n/novyy-metod-vosstanovleniya-propuschennyh-znacheniy-v-nabore-dannyh-na-primere-immunosignatur/viewer>
17. Любимцев О.В. Линейные регрессионные модели в эконометрике [Текст] : мет. пос. / О.В. Любимцев, О.Л. Любимцева. – Нижний Новгород: ННГАСУ, 2016. – 45 с.

18. Эконометрия [Текст] : учеб. / В. И. Суслов, Н. М. Ибрагимов, Л. П. Талышева, А. А. Цыплаков. – Новосибирск: СО РАН, 2005. – 744 с.
19. Бокс Дж. Анализ временных рядов, прогноз и управление [Текст] : пер.с англ / Дж. Бокс, Г. Дженкинс. – М.: Мир, 1974. – 406 с.
20. Андерсон Т. Статистический анализ временных рядов [Текст] : пер.с англ / Т. Андерсон. – М.: Мир., 1976. – 757 с.
21. Tussel F. Kalman Filterenh in R [Electronic resource] / F. Tussel. - Journal of Statistical Software. – V.39. – March 2011. – Regime on access: <https://www.jstatsoft.org/article/view/v039i02/v39i02.pdf>
22. International Financial Statement Analysis [Text] / Th. R. Robinson, H. van Greuning, E. Henry, M. A. Broihahn. - New Jersey: John Wiley & Sons, Inc, 2009. –864 p. – CFA Institute.
23. Financial Ratios - Complete List and Guide to All Financial Ratios [Electronic resource] / Corporate Finance Institute. – Regime on access: <https://corporatefinanceinstitute.com/resources/knowledge/finance/financial-ratios/>

ДОДАТОК А Лістинг програми

```

library(openxlsx) library(readxl)
library(writexl) library(shiny)
library(shinythemes) library(DT)
library(caret) library(forecast)
library(tseries) library(rugarch)
library(GMDH) library(DescTools)
library(Metrics) library(KFAS)
library(corrplot) library(lmtest)
# Define server logic
server <- function(input, output) {
  InputData <- reactive({
    infile <- input$file
    if(is.null(infile))
      return(NULL)
    read_excel(infile$datapath)})
  output$RawData <- DT::renderDataTable(
    DT::datatable({ InputData() },
      options =
list(lengthMenu=list(c(10,20,50),c('10','20','50')),pageLen
gth=30, server = TRUE,
initComplete = JS("function(settings, json) {",
"$ (this.api().table().header()).css({'background-color':
'dimgrey', 'color': 'grey43'});",",",",",",",
columnDefs=list(list(className='dt-
center',defaultContent = " ",targets="_all")),
list( targets = c(1,2), render = JS(
"function(data, type, row, meta) {",
"return type === 'display' && data.length > 1 ?",
""<span title='\" + data + "\">' + data.substr(0, 1) +
'...</span>': data;",",",",",",",
filter = "top",selection = 'multiple',style =
'bootstrap',class = 'cell-border stripe', rownames =
FALSE,
colnames = names(InputData()))))

ddata_full <- reactive({
  preProcVal <- preprocess(data %>%
dplyr::select(names(InputData()))),
method = c("knnImpute"),k = 17, knnSummary = mean)
  data_full <- predict(preProcVal, InputData(), na.action
= na.pass)
  proc <- data.frame(col = names(preProcVal$mean),
mean = preProcVal$mean, sd = preProcVal$std)
  for(i in proc$col){
    data_full[i] <-
data_full[i]*preProcVal$std[i]+preProcVal$mean[i] }
  data_full })
  output$FilledData <- DT::renderDataTable(
    DT::datatable({ ddata_full() },
      options =
list(lengthMenu=list(c(10,20,50),c('10','20','50')),pageLen
gth=30, initComplete = JS(
"function(settings, json)
{", "$ (this.api().table().header()).css({'background-color':
'dimgrey', 'color': 'grey43'});",",",",",",
columnDefs=list(list(className='dt-
center',targets="_all"))),
filter = "top",selection = 'multiple',
style = 'bootstrap',class = 'cell-border stripe',
rownames = FALSE,colnames = names(InputData()))
#-----#
Improvement <- reactive({
  ddataa <- InputData()
  ddata <- ts(ddataa[2], frequency = 4,
start=c(1975,1))
  if (input$Improvement == 1)
    { ddata1 <- ddata }
  else{
    if(input$Improvement == 2)
      {holt <- HoltWinters(ddata)

```



```

ddata1 <- ts(holt$fitted, frequency = 4)
ddata1 <- ts(ddata1[,1])
else{
  kalman <- SSMModel(ddata ~ SSMtrend(1, Q =
0.01)+ SSMseasonal(period = 4, sea.type =
"trigonometric", Q =0.01),H = 0.01)
  kalman1 <- KFS(kalman)
  ddata1<- kalman1$alphahat[,1]}
ddata1 })
Decomposition <- reactive({
  if(input$Decomposition == 1){ ddataa <-
Improvement()}
  else{
    if(input$Decomposition ==2){ ddataa <-
log(Improvement())}
    else
      {if(input$Decomposition ==3){
ddataa <- diff((Improvement()))}
      else{
        ddataa <- diff(log(Improvement())) }
      }return(ddataa)})
output$plot <- renderPlot({
  plot(Decomposition())
  title(main = "Time Series Plot")})
ACFTEST <- reactive({
  if(input$ACFTEST ==1){
    acftest <- acf(Decomposition())}
  else {acftest <- pacf(Decomposition())}
  acftest })
output$sacftest1 <- renderPlot({ ACFTEST()})
StationarityTest <- reactive({
  if(input$StationarityTest == 1){
    stationarity <- adf.test(Decomposition())}
  else
    {if(input$StationarityTest ==2){
      stationarity <- pp.test(Decomposition())}
    else
      {stationarity <- kpss.test(Decomposition())}
    }stationarity})
output$Stationarity <- renderPrint({
  StationarityTest()})
output$result1 <- renderText({
  if(StationarityTest()$p.value > 0.05){result="Нульова
гіпотеза відхиляється. Ряд НЕ СТАЦІОНАРНИЙ"}
  else{result="Нульова гіпотеза приймається. Ряд
СТАЦІОНАРНИЙ"}
  result })
output$decomposed <- renderPlot({
  ddataa <- InputData()
  ddata <- ts(ddataa[2], frequency = 4,
    start=c(1975,1))
  plot(decompose(ddata)) })
#-----#
train1 <- function(x){
  if(input$Improvement == 2){
    train <- ts(log(x)[5:(length(log(x))-4)], frequency =
4)}
  else{
    train <- ts(log(x)[1:(length(log(x))-4)], frequency = 4)}
test1 <- function(x){
  test <- tail(log(x), n = 4)}
output$ARIMA <- renderPrint(
  {auto.arima(train1(Improvement()))})
ACFTESTResid1 <- reactive({
  if(input$ACFTESTResid ==1){
    ARIMA1 <- auto.arima(train1(Improvement()))
    acftest <- acf(ARIMA1$residuals)}
  else {
    ARIMA1 <- auto.arima(train1(Improvement()))
    acftest <- acf(ARIMA1$residuals^2)}
  acftest })
output$sacftest2 <- renderPlot({
  ACFTESTResid1()})
ACFTESTResid2 <- reactive({
  if(input$ACFTESTResid ==1){

```

```

ARIMA1 <- auto.arima(train1(Improvement()))
pacftest <- pacf(ARIMA1$residuals)
else {
  ARIMA1 <- auto.arima(train1(Improvement()))
  pacftest <- pacf(ARIMA1$residuals^2)
  pacftest})
output$pacftest2 <- renderPlot({ ACFTESTResid2()})
output$TSResid <- renderPlot({
  ARIMA1 <- auto.arima(train1(Improvement()))
  tsdiag(ARIMA1)})
ARCH_effect_test <- reactive({
  if(input$ARCH_effect_test == 1){
    ARIMA1 <- auto.arima(train1(Improvement()))
    archtest <- Box.test(ARIMA1$residuals)
  } else {
    ARIMA1 <- auto.arima(train1(Improvement()))
    archtest <- Box.test(ARIMA1$residuals, type =
"Ljung-Box")
    archtest })
output$ARCHEffect <- renderPrint({
  ARCH_effect_test()})
output$result2 <- renderText({
  if(ARCH_effect_test()$p.value >
0.05){result="Нульову гіпотезу відхилено, ARCH
ефект у залишках ВІДСУТНІЙ"}
  else{result="Нульову гіпотезу прийнято, ARCH
ефект у залишках ПРИСУТНІЙ"}
  resul})
GARCHpq1 <- reactive({
  ARIMA1 <- auto.arima(train1(Improvement()))
  GARCH1 <- ugarchspec(
    variance.model = list(model = "sGARCH",
garchOrder =
c(as.numeric(input$ARCHp),as.numeric(input$ARCHq))
),
    mean.model = list(armaOrder = c(0,0), include.mean
= FALSE))
  GARCH1 <- ugarchfit(spec = GARCH1,
data=ARIMA1$residuals, solver = 'hybrid')
  return(GARCH1)})
output$GARCHpq <- renderPrint({

```

```

GARCHpq1() })
#-----#
Forecast1 <- reactive({
  if(input$Forecast == 1)
    { ARIMA1 <- auto.arima(train1(Improvement()))
      forecast1 <- predict(ARIMA1, n.ahead = 4)
      forecast_total <- as.numeric(forecast1$pred) }
  else {
    if(input$Forecast == 2){
      #2019.1
      ARIMA1 <- auto.arima(train1(Improvement()))
      arima_forecast <- predict(ARIMA1, n.ahead = 1)
      GARCH_final <- ugarchspec(
        variance.model = list(model = "sGARCH",
garchOrder =
c(as.numeric(input$ARCHp1),as.numeric(input$ARCHq
1))),
        mean.model = list(armaOrder = c(0,0),
include.mean = FALSE))
      GARCH_final1 <- ugarchfit(spec = GARCH_final,
        data=ARIMA1$residuals)
      ht1 <- 0
      for (i in 2:length(GARCH_final1@fit$coef)){
        ht1 <- ht1 +
GARCH_final1@fit$coef[i]*ARIMA1$residuals[length(t
rain1(Improvement()))-
length(GARCH_final1@fit$coef)+i]
      }
      # forecast_total <- lenght(GARCH_final1@fit$coef)
      ht1 <- ht1 + GARCH_final1@fit$coef[1]
      forecast1_1 <- arima_forecast$pred + ht1
      train1_1 <- ts(c(train1(Improvement()),forecast1_1),
frequency = 4)
      #2019.2
      ARIMA1 <- auto.arima(train1_1)
      arima_forecast <- predict(ARIMA1, n.ahead = 1)
      GARCH_final <- ugarchspec(
        variance.model = list(model = "sGARCH",
garchOrder =
c(as.numeric(input$ARCHp1),as.numeric(input$ARCHq
1))),

```

```

mean.model = list(armaOrder = c(0,0), include.mean
= FALSE))

GARCH_final1 <- ugarchfit(spec = GARCH_final,
                        data=ARIMA1$residuals)

ht2 <- 0

for (i in 2:length(GARCH_final1 @fit$coef)){
  ht2 <- ht2 +
GARCH_final1 @fit$coef[i]*ARIMA1$residuals[length(t
rain1_1)-length(GARCH_final1 @fit$coef)+i]}

ht2 <- ht2 + GARCH_final1 @fit$coef[1]

forecast1_2 <- arima_forecast$pred + ht2

train2 <- ts(c(train1_1,forecast1_2), frequency = 4)

#2019.3

ARIMA1 <- auto.arima(train2)

arima_forecast <- predict(ARIMA1, n.ahead = 1)

GARCH_final <- ugarchspec(
  variance.model = list(model = "sGARCH",
garchOrder =
c(as.numeric(input$ARCHp1),as.numeric(input$ARCHq
1))),
  mean.model = list(armaOrder = c(0,0), include.mean
= FALSE))

GARCH_final1 <- ugarchfit(spec = GARCH_final,
                        data=ARIMA1$residuals)

ht3 <- 0

for (i in 2:length(GARCH_final1 @fit$coef)){
  ht3 <- ht3 +
GARCH_final1 @fit$coef[i]*ARIMA1$residuals[length(t
rain2)-length(GARCH_final1 @fit$coef)+i]}

ht3 <- ht3 + GARCH_final1 @fit$coef[1]

forecast1_3 <- arima_forecast$pred + ht3

train3 <- ts(c(train2,forecast1_3), frequency = 4)

#2019.4

ARIMA1 <- auto.arima(train3)

arima_forecast <- predict(ARIMA1, n.ahead = 1)

GARCH_final <- ugarchspec(
  variance.model = list(model = "sGARCH",
garchOrder =
c(as.numeric(input$ARCHp1),as.numeric(input$ARCHq
1))),

```

```

mean.model = list(armaOrder = c(0,0), include.mean
= FALSE))

GARCH_final1 <- ugarchfit(spec = GARCH_final,
                        data=ARIMA1$residuals)

ht4 <- 0

for (i in 2:length(GARCH_final1 @fit$coef)){
  ht4 <- ht4 +
GARCH_final1 @fit$coef[i]*ARIMA1$residuals[length(t
rain3)-length(GARCH_final1 @fit$coef)+i]}

}

ht4 <- ht4 + GARCH_final1 @fit$coef[1]

forecast1_4 <- arima_forecast$pred + ht4

forecast_total <- c(forecast1_1, forecast1_2,
forecast1_3, forecast1_4) }

else {if( input$Forecast == 3){
  forecast1 <- fcast(train1(Improvement()), weight =
0.7, method = "GMDH", f.number = 4, layer = 1)

  forecast_total <- as.numeric(forecast1$mean) }

else{if (input$Forecast ==4){
  nnar <- nnetar(train1(Improvement()))

  forecast1 <- forecast(nnar, h = 4)

  forecast_total<- as.numeric(forecast1$mean) }

else{ ARIMA1 <-
auto.arima(train1(Improvement()))

  nnar <- nnetar(ARIMA1$residuals)

  arima_forecast <- predict(ARIMA1, n.ahead = 4)

  forecast1 <- forecast(nnar, h = 4)

  forecast_total <-as.numeric(arima_forecast$pred +
forecast1$mean)}}} forecast_total))

output$Forecast1 <- renderPrint({
  # data.frame(Forecast1(), test_data, row.names
=list("2019 Q1", "2019 Q2", "2019 Q3", "2019 Q4"))
  Forecast1() })

output$RealTest <- renderPrint({
  as.numeric(tail(log(Improvement()), n = 4)) })

output$Forecast_results <- renderPrint({
  cbind(c("MSE", "MAE", "MAPE",
"TheilU"),c(mse(Forecast1(), test1(Improvement()))),
mae(Forecast1(), test1(Improvement())),
mape(Forecast1(), test1(Improvement())),
TheilU(Forecast1(), test1(Improvement())))) })

```

```
#-----#

Regressors <- reactive({

  data_full_1 <- ddata_full()[c(match(input$Regressors,
colnames1)+1)]

  data_full_1 })

output$Corplot <- renderPlot({

  corplot(cor(Regressors())) })

output$Model <- renderPrint({

  xnames <- colnames(Regressors())

  model1 <-
as.formula(paste((colnames(Regressors()))[1], "~"
,paste(xnames[2:length(xnames)], collapse = " + ")))

  model1 })

output$Model11 <- renderPrint({

  xnames1 <- colnames(Regressors())

  model11 <-
as.formula(paste((colnames(Regressors()))[1], "~"
,paste(xnames1[2:length(xnames1)], collapse = " + ")))

  model1_1 <- lm(model11, data = ddata_full())

  model1_1 })

output$Assess <- renderPrint(({

  xnames1 <- colnames(Regressors())

  model11 <-
as.formula(paste((colnames(Regressors()))[1], "~"
,paste(xnames1[2:length(xnames1)], collapse = " + ")))

  model1_1 <- lm(model11, data = ddata_full())

  return (cbind(c("R squared", "Adjusted R squared",
"Dw" , "AIC", "BIC"),
c(summary(model1_1)$r.squared,summary(model1_1)$a
dj.r.squared, dwtest(model1_1)$Statistic,

      AIC(model1_1),

      BIC(model1_1))))))

output$AssessForecast <- renderPrint(({

  xnames1 <- colnames(Regressors())

  model11 <-
as.formula(paste((colnames(Regressors()))[1], "~"
,paste(xnames1[2:length(xnames1)], collapse = " + ")))

  model1_1 <- lm(model11, data = ddata_full())

  test2 <- as.numeric(c(ddata_full()[nrow(ddata_full())-
3,2], ddata_full()[nrow(ddata_full())-
2,2],ddata_full()[nrow(ddata_full())-
1,2],ddata_full()[nrow(ddata_full()),2]))
```

```
predict_val <- predict(model1_1, newdata =
tail(ddata_full()[,-1], n = 4 ))

  return (cbind(c("MSE", "MAE", "MAPE" , "TheilU"),
c(mse(as.numeric(predict_val), test2),
mae(as.numeric(predict_val),
test2),mape(as.numeric(predict_val), test2),
TheilU(as.numeric(predict_val), test2) )))))

ACFTEST2 <- reactive({

  xnames1 <- colnames(Regressors())

  model11 <-
as.formula(paste((colnames(Regressors()))[1], "~"
,paste(xnames1[2:length(xnames1)], collapse = " + ")))

  model1_1 <- lm(model11, data = ddata_full())

  if(input$ACFTEST2 ==1){

    acftest <- acf(model1_1$residuals)}

  else { acftest <- pacf(model1_1$residuals) }

  acftest })

output$acftest3 <- renderPlot({

  ACFTEST2()})

output$residplot <- renderPlot({

  xnames1 <- colnames(Regressors())

  model11 <-
as.formula(paste((colnames(Regressors()))[1], "~"
,paste(xnames1[2:length(xnames1)], collapse = " + ")))

  model1_1 <- lm(model11, data = ddata_full())

  ts.plot(model1_1$residuals) })

StationarityTest2 <- reactive({

  xnames1 <- colnames(Regressors())

  model11 <-
as.formula(paste((colnames(Regressors()))[1], "~"
,paste(xnames1[2:length(xnames1)], collapse = " + ")))

  model1_1 <- lm(model11, data = ddata_full())

  if(input$StationarityTest2 == 1){

    stationarity <- adf.test(model1_1$residuals)}

  else{ if(input$StationarityTest2 ==2){

    stationarity <- pp.test(model1_1$residuals)}

    else{

      stationarity <- kpss.test(model1_1$residuals, null =
"Trend")}}

  stationarity })

output$Stationarity2 <- renderPrint({
```

```

StationarityTest2()))
output$result4 <- renderText({
  if(StationarityTest2())$p.value > 0.05){result="Нульова
гіпотеза відхиляється. Ряд НЕ СТАЦІОНАРНИЙ"}
  else{result="Нульова гіпотеза приймається. Ряд
СТАЦІОНАРНИЙ"}
  result})
output$ARIMA2 <- renderPrint(
  {xnames1 <- colnames(Regressors())
  model11 <-
as.formula(paste((colnames(Regressors()))[1], "~"
,paste(xnames1[2:length(xnames1)], collapse = " + "))
  model1_1 <- lm(model11, data = ddata_full())
  auto.arima(model1_1$residuals)})
output$Forecast2 <- renderPrint({
  h <- 0
  xnames1 <- colnames(Regressors())
  xnames2 <- c(input$Regressors)
  xnames3 <- xnames2[names(xnames2)!="1"]
  model11 <-
as.formula(paste((colnames(Regressors()))[1], "~"
,paste(xnames1[2:length(xnames1)], collapse = " + "))

```

```

model1_1 <- lm(model11, data = ddata_full())
#arima_model <- auto.arima(model1_1$residuals)
arima_model <- arima(model1_1$residuals, order =
c(4,1,1))
for( i in 2:length(model1_1$coefficients)){
  h<-h + model1_1$coefficients[i]*Regressors()[180, i]}
h <- h + model1_1$coefficients[1]
h1 <- 0
for( i in 1: length(arima_model$coef)){
  h1 <- h1 + arima_model$coef[i] *
model1_1$residuals[length(ddata_full()) -
length(arima_model$coef)+i]}
h2 <- h + h1
cbind(c("Predicted value", "Real value", "MAPE",
"TheilU"),c(h2, tail(ddata_full()[,2], n =1 ),
(abs(tail(ddata_full()[,2], n =1 )-
h2)/abs(tail(ddata_full()[,2], n =1 ))) , TheilU(h2,
tail(ddata_full()[,2], n =1 ))) )})
# Run the application
shinyApp(ui = ui, server = server)

```

ДОДАТОК Б Ілюстративний матеріал для доповіді

МОДЕЛЮВАННЯ ТА ПРОГНОЗУВАННЯ ФІНАНСОВОЇ СТІЙКОСТІ КОМПАНІЇ З ВИКОРИСТАННЯМ МЕТОДІВ ІАД

ВИКОНАЛА:

студентка групи КА-64
Черниш Злата

КЕРІВНИК:

д.т.н., доцент кафедри ММСА
Кузнєцова Н.В.

ОБ'ЄКТ ДОСЛІДЖЕННЯ:

Вибірка даних, взятих із
звітів про фінансові результати та
фінансові коефіцієнти, що
потребують подальшого аналізу із
виявленням закономірностей та
взаємозалежностей

ПРЕДМЕТ ДОСЛІДЖЕННЯ:

Методи та моделі прогнозування
часових рядів:

- АР, АРКС, АРІКС, АРУГ,
- лінійної регресії,
- на основі нейронних мереж,
- на основі МГУА,
- згладжування та фільтрація
часових рядів.

МЕТА ДОСЛІДЖЕННЯ:

РОЗРОБКА ТА
РЕАЛІЗАЦІЯ
ІНФОРМАЦІЙНОЇ
СИСТЕМИ ДЛЯ
МОДЕЛЮВАННЯ ТА
ПРОГНОЗУВАННЯ
ЕКОНОМІЧНИХ ТА
ФІНАНСОВИХ ЯВИЩ
ТА ПОКАЗНИКІВ

ФІНАНСОВЕ**ПРОГНОЗУВАННЯ –**

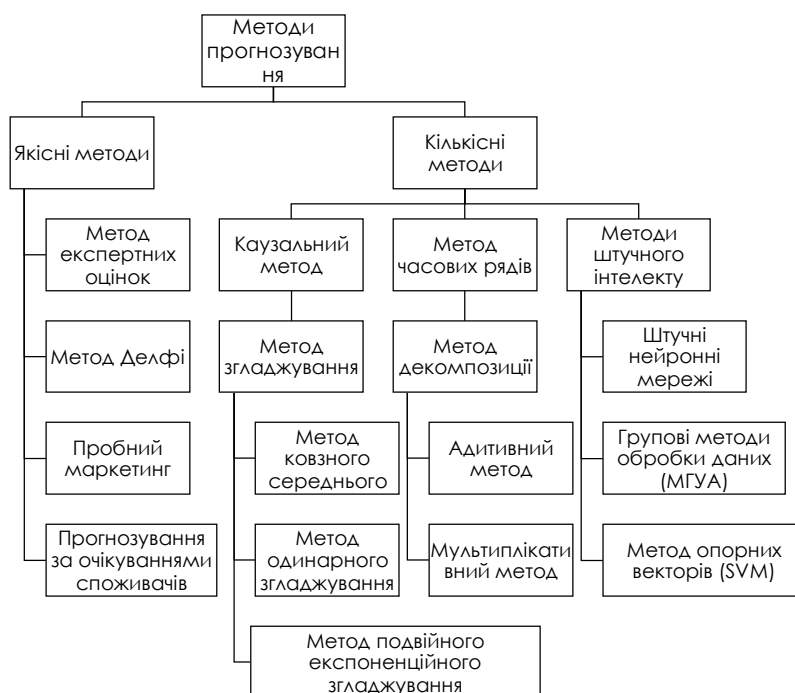
процес розробки та реалізації прогнозів, метою яких є ефективна організація діяльності підприємства, передбачення наслідків, прийняття певних рішень.

Мета:

оцінка очікуваного об'єму фінансових ресурсів, пошук джерел формування та напрямів ефективного використання фінансових ресурсів, формування рекомендацій щодо раціональної фінансової стратегії та тактики менеджменту компанії

3

Методи прогнозування



4

МОДЕЛЬ З ТРЕНДОМ АРИКС(p,d,q)

Інтегрована модель авторегресії-ковзного середнього (модель Бокса-Дженкінса, ARIMA)

Після процедури взяття послідовних різниць можна привести до виду АРКС.

Формула:

$$\Delta^d y_t = \alpha_1 \Delta^d y_{t-1} + \alpha_2 \Delta^d y_{t-2} + \dots + \alpha_p \Delta^d y_{t-p} + \varepsilon_t - \beta_1 \varepsilon_{t-1} - \dots - \beta_q \varepsilon_{t-q}$$

$y = y_t, t \geq 1$ – певний процес

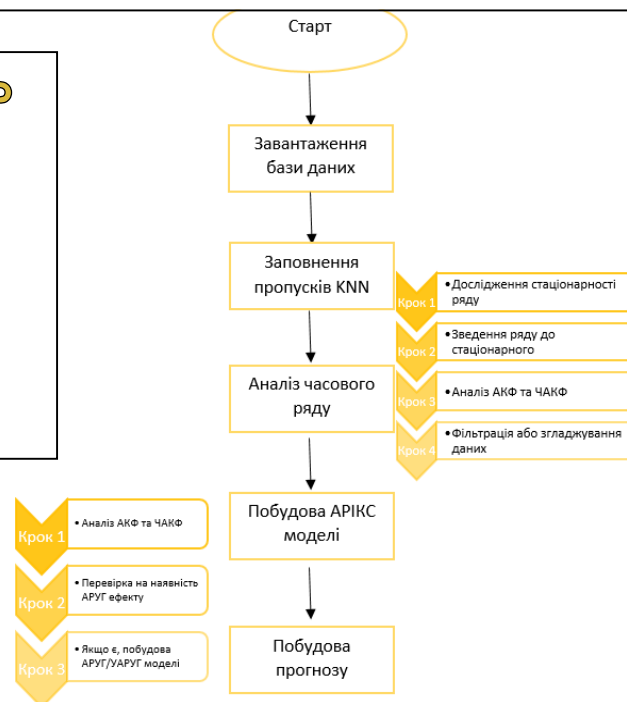
Δ^d – оператор різності часового ряду порядку d,

α, β – параметри моделі

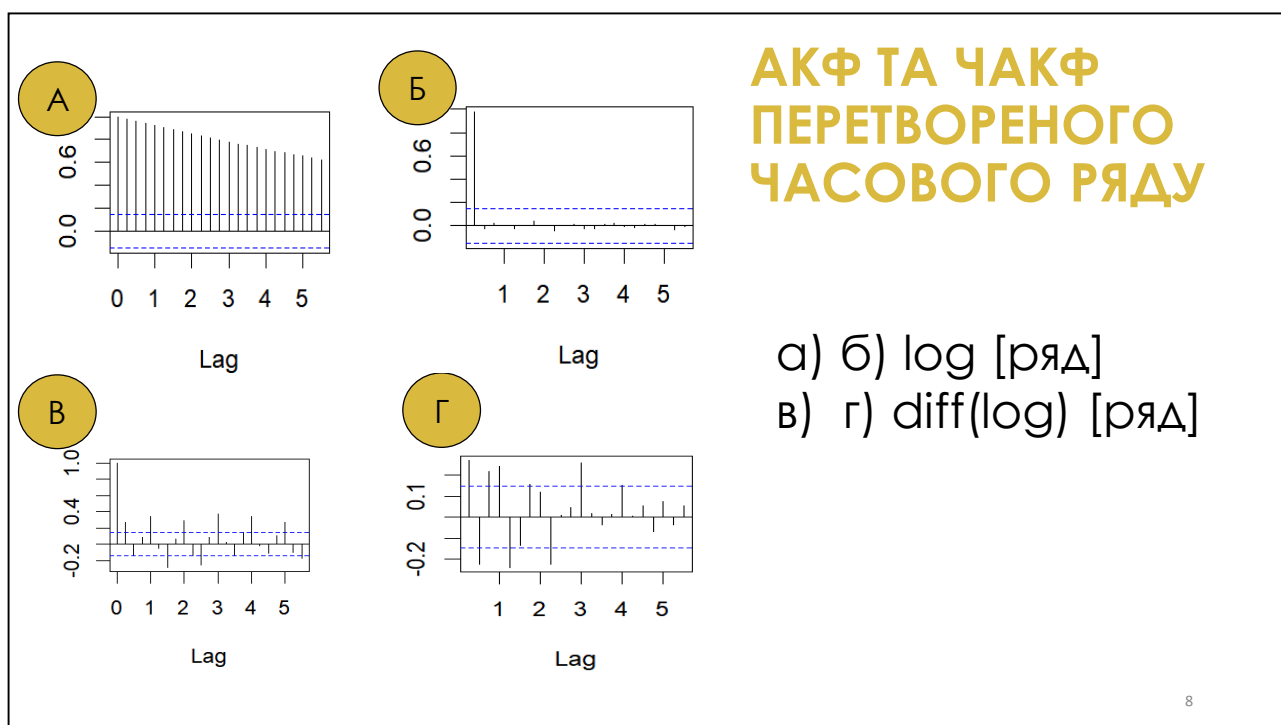
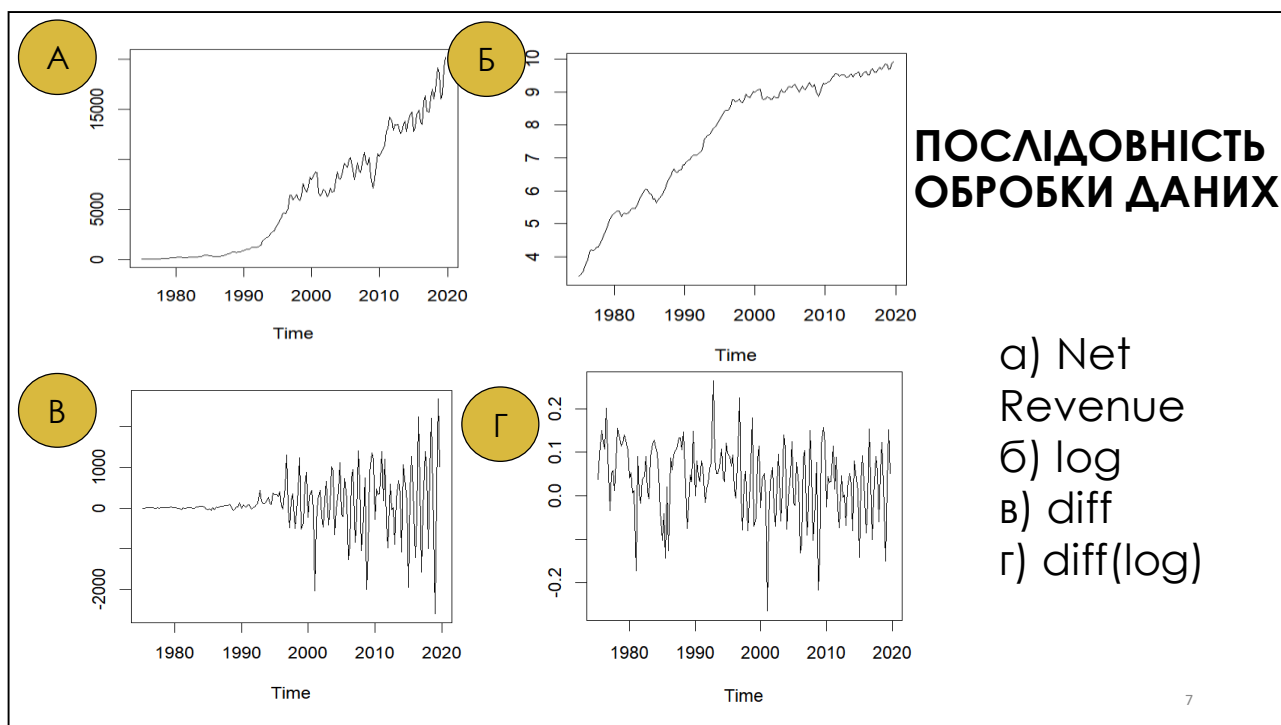
ε_t – білий шум

5

ПОСЛІДОВНІСТЬ РЕАЛІЗАЦІЇ ПОБУДОВИ МОДЕЛІ АРИКС(p,d,q)



6



СЕЗОННА МОДЕЛЬ

SARIMA(0,2,1)(2,0,1)[4]

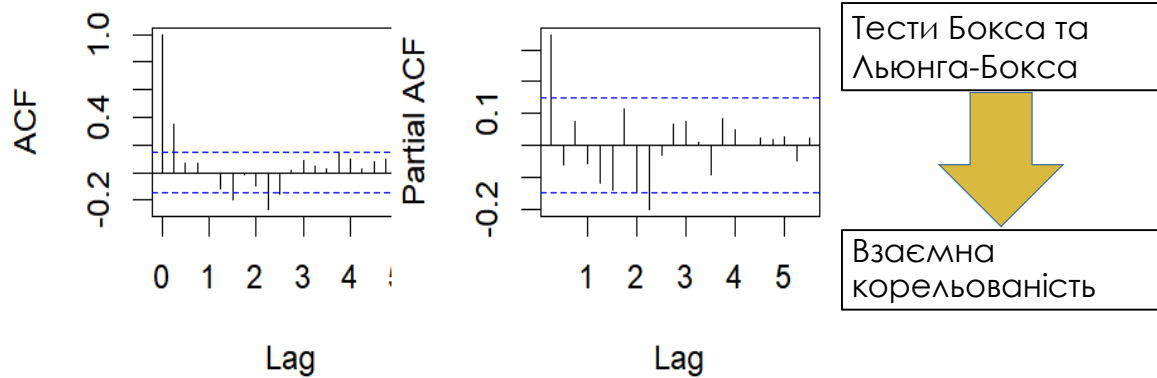
ma1 sar1 sar2 sma1
-0.9849 0.9005 0.0506 -0.7649

КРИТЕРІЇ АДЕКВАТНОСТІ

$AIC = -406.5$ $AICc = -406.14$ $BIC = -390.7$.

9

Аналіз залишків сезонної моделі



10

МОДЕЛЬ АРУГ(p,q)

Авторегресійна умовно гетероскедастична модель (АРУГ)

Застосування: моделювання волатильності доходності фінансових активів

Маємо:

Залишки моделі АРІКС - $\varepsilon_t \sim N(0, \sigma_t^2)$.

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^p \alpha_i y_{t-i}^2$$

де p – максимальний порядок АРУГ,

σ_t^2 – умовна дисперсія,

y_{t-i} – попередні значення,

α_0, α_i – параметри моделі, $\alpha_0 > 0, \alpha_i \geq 0, i = 1, \dots, p$

11

ПОРІВНЯННЯ АРУГ/УАРУГ МОДЕЛЕЙ

Модель	AIC	BIC
АРУГ (1)	-2.4089	-2.3729
АРУГ (2)	-2.4106	-2.3566
УАРУГ(1,1)	-2.3968	-2.3427
УАРУГ(1,2)	-2.3855	-2.3134
УАРУГ(2,1)	-2.3854	-2.3134
УАРУГ(2,2)	-2.3741	-2.2841

Optimal Parameters

	Estimate	Std. Error	t value	Pr(> t)
omega	0.004598	0.000747	6.15443	0.00000
alpha1	0.018598	0.113494	0.16386	0.86984
alpha2	0.101433	0.088022	1.15236	0.24917

12

13

МГГА (ГМДН)

14

НЕЙРОННІ АВТОРЕГРЕСІЙНІ МОДЕЛІ (NNAR)

Модель	Спрогнозовані значення	MSE	MAE	MAPE	THEILU
NNAR	9.79478	0.008741967	0.09063373124	0.0092557203	0.00954968
	9.77772	11462989	87144	9153677	431360333
	9.79289				
	9.79755				
Хольт-Вінтерс NNAR	9.838072	0.0036732939	0.05084543181	0.0051721848	0.00616538
	9.825942	2362531	29994	6559634	611760692
	9.825962				
	9.831290				
Калман NNAR	9.779976	0.0004515699	0.01632996610	0.0016716063	0.00217419
	9.777926	06136031	52245	6155147	224014271
	9.772345				
	9.765046				

15

ГІБРИДНА МОДЕЛЬ ARIMA+ NNAR

Модель	Спрогнозовані значення	MSE	MAE	MAPE	THEILU
SARIMA(0,2,1)(2,0,1)[4] + NNAR	9.688160	0.004500317	0.05144639652	0.005256050	0.00687288
	9.745717	34377418	79743	81154131	854625651
	9.815720				
	9.792831				
Хольт-Вінтерс SARIMA(1,2,2)(0,0,1)[4]+ NNAR	9.832237	0.014071091	0.10847055422	0.010953885	0.01199001
	9.911051	4788757	6279	2377955	50686092
	9.927265				
	9.902652				
Калман SARIMA+NNAR	9.777094	3.215726871	0.00454298723	0.0004636459	0.00057915
	9.782053	48903e-05	00593	43741009	7892207037
	9.795078				
	9.811141				

16



У якості регресорів виступатимуть наступні показники:

- X_1 – Gross Margin,
- X_2 –ICR
- X_3 –ROA
- X_4 –ROE
- X_5 –ATR
- X_6 –Price

Оцінки адекватності моделі

"R squared"	"0.823688768537593"
"Adjusted R squared"	"0.817573928140053"
DW "Dw"	"0.675801362443861"
"AIC"	"3324.80957056574"
"BIC"	"3350.35322537286"

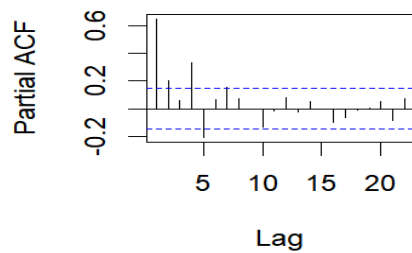
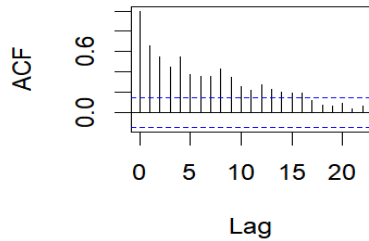
Оцінки якості прогнозу моделі

	[,1]	[,2]
[1,] "MSE"	"941829.557998272"	
[2,] "MAE"	"800.309020313319"	
[3,] "MAPE"	"0.0440706910945356"	
[4,] "TheilU"	"0.0533471196569216"	

$$Y = 8771.794X_1 + 10.165X_2 - 166.813X_3 + 136.720X_4 - 32.718X_5 + 206.680X_6 + 2755.078 + \varepsilon.$$

18

Аналіз залишків моделі лінійної регресії



ADF тест

РЯД СТАЦІОНАРНИЙ

PP тест

KPSS Тест

Є ТРЕНД

19

ХАРАКТЕРИСТИКА МОДЕЛІ АРІКС(3,1,3)

```

Coefficients:
      ar1      ar2      ar3      ma1
    -0.7962 -0.4537 -0.4484  0.3943
s.e.    0.1290  0.0917  0.0692  0.1361

sigma^2 estimated as 2783157:  log likelihood=-158
AIC=3170.95  AICc=3171.3  BIC=3186.89

```

ПРОГНОЗ

20592.47

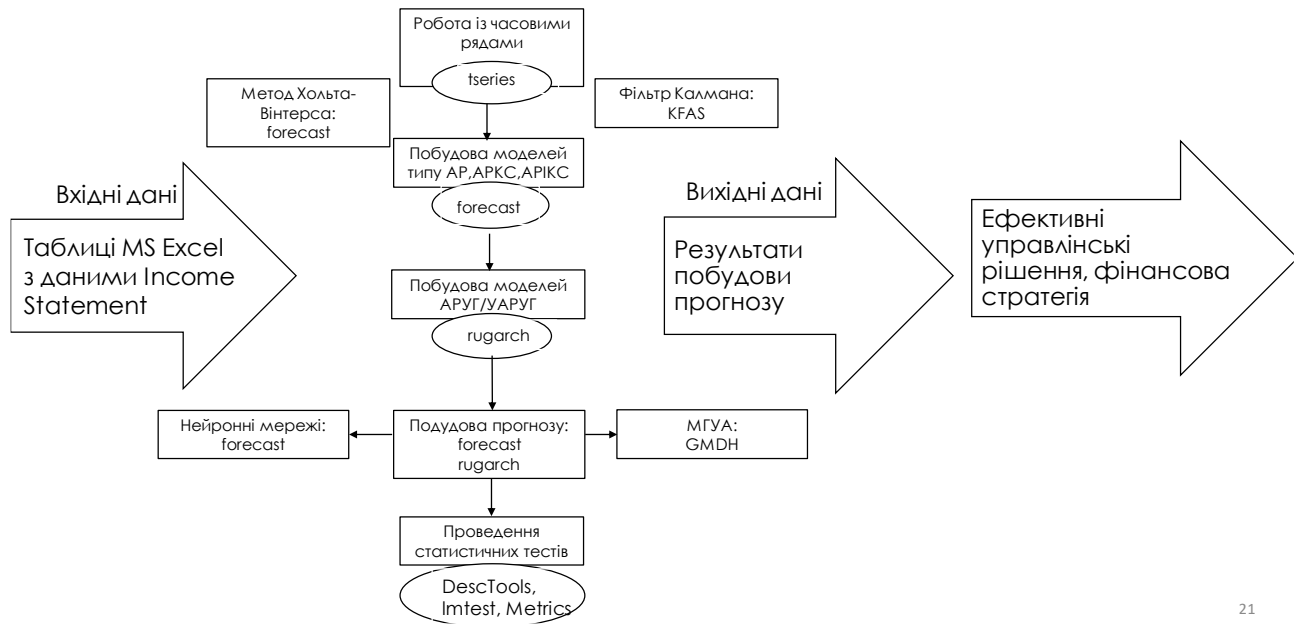
РЕАЛЬНЕ ЗНАЧЕННЯ 20209

MAPE = **0.01897498**,

TheilU = **0.01862163**

20

ЕЛЕМЕНТИ ІНФОРМАЦІЙНОЇ СИСТЕМИ



21

ВИСНОВКИ

Розглянуто основні методи і моделі для задачі аналізу та прогнозування фінансових показників та процесів, а саме:

- авторегресії, авторегресії – інтегрованого ковзного середнього;
- авторегресії з умовною гетероседастичністю;
- на основі методу групового урахування аргументу;
- на основі нейронних мереж;
- лінійної регресії.

Опрацьовано та реалізовано методи фільтрації та згладжування даних – Хольта-Вінтерса та фільтр Калмана.

Розроблено інформаційну систему, за допомогою якої виконано моделювання та прогнозування фінансової діяльності компанії Intel Corporation.

22